Introduction to Artificial Intelligence

COMP307

Uncertainty and Probability 5: 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

推论

Inference

概率推理系统

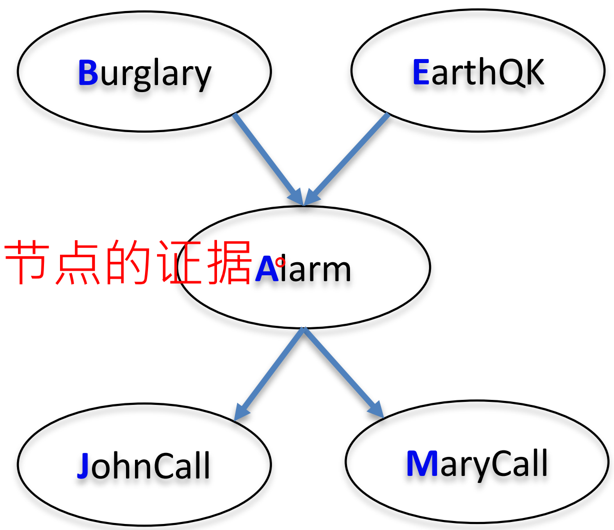
- The basic task for any probabilistic inference system is to compute the posterior probability distribution for a set of **query nodes**, given values for some **evidence nodes**.
  查询节点
  - What is P(Burglary=true), if we know that (Alarm=true)?

- This task is called belief updating or probabilistic inference.

  信念更新或概率推理

- Inference in Bayesian networks is very flexible, as evidence can be entered about any node while beliefs in any other nodes are updated.

- Major classes of inference algorithms
  - exact (Inference by Enumeration)
  - approximate



贝叶斯网络推断非常灵活，
因为可以在任何其他节点中的belief信度中输入任何节点的证据

•主要推理算法类 -
 精确（枚举推断）
 - 近似值

Inference

- For some networks, exact inference becomes computationally infeasible, in which case approximate inference algorithms must be used.

- In general, both exact and approximate inference are theoretically computationally complex (specifically, NP hard).

- Speed of inference depends on factors such as the structure of the network, including
  - how highly connected it is
  - the locations of evidence
  - query nodes

Inference by Enumeration

- Problem (**<u>capital letter = variable, lowercase = value</u>**):
  - Given evidence nodes: $e_1, e_2, \dots, e_n$
    - e.g., 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: $H_1, H_2, \dots, H_m$
  - $P(Q|e_1, \dots, e_n)$?

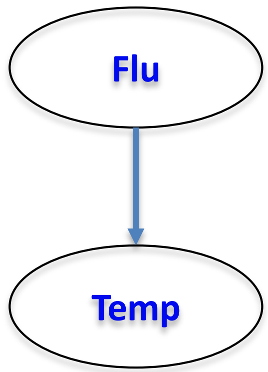
- Use the 3 probability rules
  - Product rule
  - Sum rule
  - Normalisation rule

Inference by Enumeration

- A simple example: 2-node network
- How likely is the flu, given high temperature?
  - Evidence node: $Temp = h$ (**high temperature**)
  - Query node: Flu
  - NO hidden node
- $P(Flu|h) = \alpha * P(Flu, h)$
- $P(Flu, h) = P(Flu)P(h|Flu)$
  - $P(f)P(h|f) = 0.05 * 0.9 = 0.045$
  - $P(\neg f)P(h|\neg f) = 0.95 * 0.2 = 0.19$
- Normalisation
  - $P(f|h) = \frac{0.045}{0.045+0.19}$
  - $P(\neg f|h) = \frac{0.19}{0.045+0.19}$

Flu	Probability
f	$P(Flu = f) = 0.05$

Flu	Temp	Probability
f	h	$P(Temp = h\,|Flu = f) = 0.9$
$\neg f$	h	$P(Temp = h\,|Flu = \neg f) = 0.2$



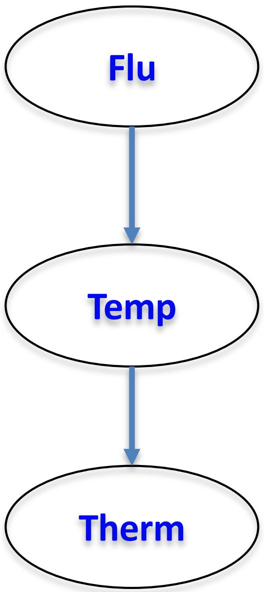
Inference by Enumeration

- Advanced example: 3-node network
  - Thermometer: 5% false negative, 15% false positive reading
- How likely is the flu, given positive thermometer reading?
  - Evidence node: $Therm = p$ (**positive, or high reading**)
  - Query node: Flu
  - Hidden node: $Temp$

Flu	Probability
f	$P(Flu = f) = 0.05$

Flu	**Temp**	Probability
f	h	$P(Temp = h \mid Flu = f) = 0.9$
$\neg f$	h	$P(Temp = h \mid Flu = \neg f) = 0.2$

Temp	**Therm**	Probability
h	t	$P(Therm = t \mid Temp = h) = 0.95$
$\neg h$	t	$P(Therm = t \mid Temp = \neg h) = 0.15$



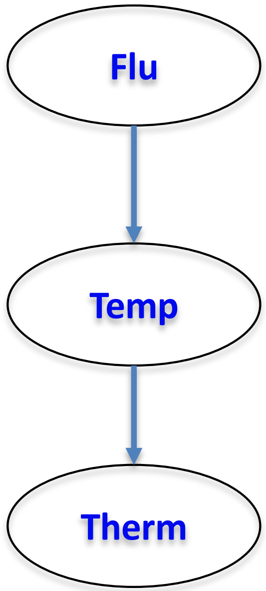
Inference by Enumeration

- $P(Flu|p) = \alpha * P(Flu, p)$
  - α is the denominator, NO need to calculate directly
- $P(Flu, p) = P(Flu, h, p) + P(Flu, \neg h, p)$
- $P(Flu, p) = P(Flu)P(h|Flu)P(p|h) + P(Flu)P(\neg h|Flu)P(p|\neg h)$
  - $P(f, p) = P(f)P(h|f)P(p|h) + P(f)P(\neg h|f)P(p|\neg h)$
  - $P(\neg f, p) = P(\neg f)P(h|\neg f)P(p|h) + P(f)P(\neg h|\neg f)P(p|\neg h)$
- Normalisation
  - $P(f, p) = \frac{P(f,p)}{P(f,p)+P(\neg f,p)}$

Flu	Probability
f	$P(Flu = f) = 0.05$

Flu	Temp	Probability
f	h	$P(Temp = h\ |Flu = f) = 0.9$
$\neg f$	h	$P(Temp = h\ |Flu = \neg f) = 0.2$

Temp	Therm	Probability
h	p	$P(Therm = p\ |Temp = h) = 0.95$
$\neg h$	p	$P(Therm = p\ |Temp = \neg h) = 0.15$



Inference by Enumeration

$$P(Q|e_1,\ldots,e_n) = \alpha * P(Q,e_1,\ldots,e_n)$$

- For the enumerator: include the hidden nodes

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

H:Hidden

- Use factorisation of the network

$$P(Q,e_1,\ldots,e_n,H_1,\ldots,H_m)$$
$$= P(Q|parents(Q)) * P(e_1|parents(E_1)) * \cdots * P(H_m|parents(H_m))$$

- How many calculations? Assume all binary variables

Inference by Enumeration

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

- For the enumerator: include the hidden nodes

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

- Use factorisation of the network

$$P(Q,e_1,\dots,e_n,H_1,\dots,H_m)$$
$$= P(Q|parents(Q)) * P(e_1|parents(E_1)) * \cdots * P(H_m|parents(H_m))$$

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

- Calculate for all the Q values
  - $\alpha = \frac{1}{\sum_Q P(Q,e_1,\dots,e_n)}$

Variable Elimination Algorithm

- Directly calculating all the joint probabilities can be time consuming
  - $P(Q, e_1, \ldots, e_n) = \sum_{H_1,\ldots,H_m} P(Q, e_1, \ldots, e_n, H_1, \ldots, H_m)$
  - $(n + m)2^{m+1}$ calculations
  - Exponential to the number of hidden variables
  - Can be very slow in large Bayesian networks
- Variable Elimination
  - Many duplicate multiplications between conditional probabilities
  - Calculate once and save for later use

Join Factors

- First basic operation: joining factors
  - There are a lot of things (tables) we can encounter in elimination: all called factors
  - Initial factors are local CPTs (one per node, fixing the evidence nodes)
- Combining factors:
  - just like a database join
  - get all factors over the joining variable
  - Build a new factor over the union of variables involved
- Example of Join on R
  - Computation for each entry: point wise products

$$f_1(A) = P(j|A) = \begin{pmatrix} P(j|a) \\ P(j|\neg a) \end{pmatrix} \qquad f_2(A) = P(m|A) = \begin{pmatrix} P(m|a) \\ P(m|\neg a) \end{pmatrix}$$

$$f_1 \; join \; f_2(A) = \begin{pmatrix} P(j|a) * P(m|a) \\ P(j|\neg a) * P(m|\neg a) \end{pmatrix}$$

Eliminate

- Second basic operation: marginalisation
- Take a factor and sum out a variable
  - Shrinks a factor to a smaller one
  - A projection operation

取一个因子求一个变量的和
–将因子缩小为较小的因子
–投影操作

R, T, L

+r	+t	+l	0.024
+r	+t	-l	0.056
+r	-t	+l	0.002
+r	-t	-l	0.018
-r	+t	+l	0.027
-r	+t	-l	0.063
-r	-t	+l	0.081
-r	-t	-l	0.729

Sum out R

T, L

$P(T, L)$

+t	+l	0.051
+t	-l	0.119
-t	+l	0.083
-t	-l	0.747

Sum out T

L

$P(L)$

+l	0.134
-l	0.886

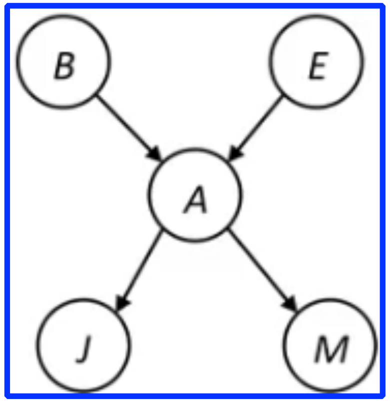
Variable Elimination Algorithm

- 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** sum out/eliminate 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

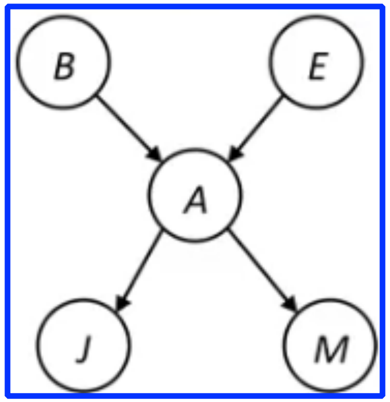
Variable Elimination Algorithm

- The alarm network example
- How likely there was a burglary, if both John and Mary called?
  - Evidence nodes: j, m
  - Query node: B
  - Hidden nodes: A, E
- $P(B|j,m) = \alpha * P(B,j,m)$
- $P(B|j,m) = \alpha * \sum_{A,E} P(B,A,E,j,m)$
- $P(B|j,m) = \alpha * \sum_{A,E} P(B)P(E)P(A|B,E)P(j|A)P(m|A)$
- **No variable elimination**, how many probability multiplications?



Variable Elimination Algorithm

- $P(B|j,m) = \alpha * \sum_{A,E} P(B)P(E)P(A|B,E)P(j|A)P(m|A)$
- How many probability multiplications?
  - Need to calculate 2 probabilities: $P(b|j,m)$ and $P(\neg b|j,m)$
  - For calculating each probability, need to sum up 2*2=4 terms
    - $(A = a, E = e)$, $(A = a, E = \neg e)$, $(A = \neg a, E = e)$, $(A = \neg a, E = \neg e)$
  - To calculate each term, there are 4 multiplications
    - $P(B) * P(e) * P(a|B,e) * P(j|a) * P(m|a)$
    - $P(B) * P(\neg e) * P(a|B,\neg e) * P(j|a) * P(m|a)$
    - $P(B) * P(e) * P(\neg a|B,e) * P(j|\neg a) * P(m|\neg a)$
    - $P(B) * P(\neg e) * P(\neg a|B,\neg e) * P(j|\neg a) * P(m|\neg a)$

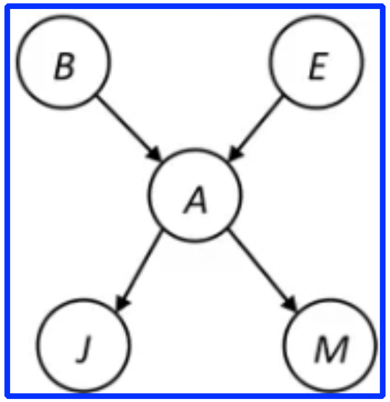


- In total: 2*4*4=32 multiplications

Variable Elimination Algorithm

- The alarm network example
  - Evidence nodes: j, m
  - Query node: B
  - Hidden nodes: A, E
- $P(B|j,m) = \alpha * P(B,j,m)$
- $P(B|j,m) = \alpha * \sum_{A,E} P(B,A,E,j,m)$
- $P(B|j,m) = \alpha * \sum_{A,E} P(B)P(E)P(A|B,E)P(j|A)P(m|A)$
- Variable Elimination
- Order: J, M, A, E, B

- How many multiplications?



Variable Elimination Algorithm

- $P(B|j,m) = \alpha * P(B) \sum_E P(E) \sum_A P(A|B,E) P(j|A) P(m|A)$
- $P(B|j,m) = \alpha * f_1(B) \sum_E f_2(E) \sum_A f_3(A,B,E) f_4(A) f_5(A)$
- f_1, f_2, f_3, f_4, f_5 are initial factors
- Step 1: sum up A
  - $f_6(B,E) = \sum_A f_3(A,B,E) f_4(A) f_5(A)$ 4*2*2 multiplications
  - $P(B|j,m) = \alpha * f_1(B) \sum_E f_2(E) f_6(B,E)$
- Step 2: sum up E
  - $f_7(B) = \sum_E f_2(E) f_6(B,E)$ 2*2*1 multiplications
  - $P(B|j,m) = \alpha * f_1(B) f_7(B)$ 2*1 multiplications
- 16+4+2=22 multiplications

Summary

- Inference
  - Enumeration
  - Variable Elimination

- Reading: Chapter 14