Inference in Bayesian Networks

Notions and Algorithms

Probabilistic inference algorithms

- Conditional independence: basis for compact representation and faster probabilistic inference
- Exact inference algorithms:
  - 'Naive' Bayesian inference
  - Judea Pearl's algorithm
  - Algorithm by Spiegelhalter & Lauritzen – implementation version by Finn Jensen

Notation: V_i's denote variables and vertices (nodes) at the same time

Naive inference

- We can perform inference by using:
  - chain rule:

$$P(V_1, V_2, \ldots, V_n) = \prod_{i=1}^{n-1} P(V_i \mid V_{i+1}, \ldots, V_n) P(V_n)$$

  - marginalisation:

$$P(V_j) = \sum_{V_1, \ldots, V_{j-1}, V_{j+1}, V_n} P(V_1, \ldots, V_n)$$

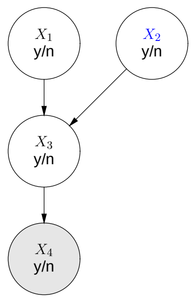
  - Bayes' rule (with evidence E):

$$P(V_j \mid E) = \frac{P(E \mid V_j) P(V_j)}{P(E)}$$

Naive inference (cont)

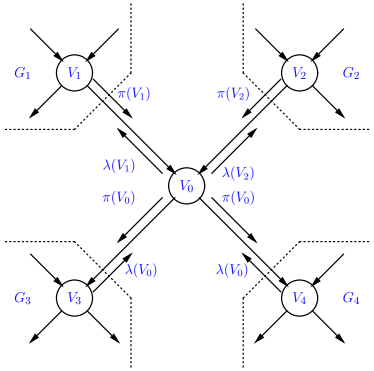
- Complexity: exponential in the number of nodes
- Becomes computationally feasible when independence among variables is explored

$$P(V(G)) = \prod_{i=1}^{n} P(V_i \mid \text{parents}_G(V_i))$$



$P(x_4 \mid x_3) = 0.4$
$P(x_4 \mid \neg x_3) = 0.1$
$P(x_3 \mid x_1, x_2) = 0.3$
$P(x_3 \mid \neg x_1, x_2) = 0.5$
$P(x_3 \mid x_1, \neg x_2) = 0.7$
$P(x_3 \mid \neg x_1, \neg x_2) = 0.9$
$P(x_1) = 0.6$
$P(x_2) = 0.2$

Basic idea of Pearl's algorithm

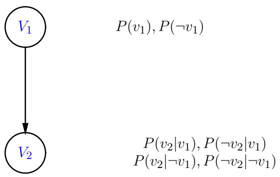


- Object-oriented approach: vertices are objects, which have local information and carry out local computations
- Updating of probability distribution by message passing: arcs are communication channels

Algorithm steps

- At each interaction the probability of a node V_i is *locally* updated
- For this, three types of parameters are used:
  - $\pi(V_i)$: messages V_i received from its parents
  - $\lambda(V_i)$: messages V_i received from its children
  - relevant CPTs values
- Then, π and λ messages are sent from V_i to its neighbours
- In short, local computation steps are:
  - Updating probabilities on node V_i
  - Propagation of π and λ messages from V_i

Probabilistic inference as message passing

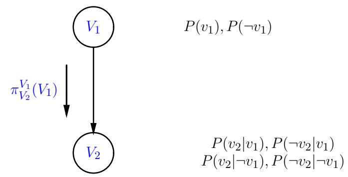


- Vertex V_1: known $P(v_1)$ and $P(\neg v_1)$
- Vertex V_2: known $P(V_2|V_1)$
- It holds that:

$$\begin{aligned} P(v_2) &= P(v_2|v_1)P(v_1) + P(v_2|\neg v_1)P(\neg v_1) \\ P(\neg v_2) &= P(\neg v_2|v_1)P(v_1) + P(\neg v_2|\neg v_1)P(\neg v_1) \end{aligned}$$

V_2 needs $P(V_1)$ which is sent from V_1 to V_2 as $\pi^{V_1}_{V_2}(V_1)$

Message passing: causal parameter $\pi^{V_i}_{V_j}$



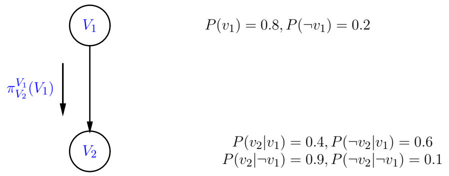
It holds that: $\pi^{V_1}_{V_2}(v_1) = P(v_1)$ and $\pi^{V_1}_{V_2}(\neg v_1) = P(\neg v_1)$

Local computation in V_2:

$$\begin{aligned} P(v_2) &= P(v_2|v_1)\pi^{V_1}_{V_2}(v_1) + P(v_2|\neg v_1)\pi^{V_1}_{V_2}(\neg v_1) \\ P(\neg v_2) &= P(\neg v_2|v_1)\pi^{V_1}_{V_2}(v_1) + P(\neg v_2|\neg v_1)\pi^{V_1}_{V_2}(\neg v_1) \end{aligned}$$

$\pi^{V_i}_{V_j}$ is called a causal parameter

Example: causal parameter $\pi_{V_j}^{V_i}$

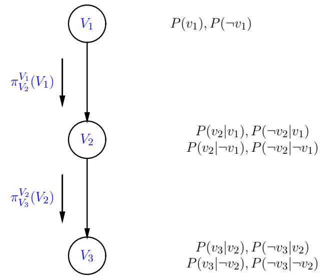


We have: $\pi_{V_2}^{V_1}(v_1) = P(v_1) = 0.8$ and $\pi_{V_2}^{V_1}(\neg v_1) = P(\neg v_1) = 0.2$

Local computation in V_2:

$$\begin{aligned} P(v_2) &= P(v_2|v_1)\pi_{V_2}^{V_1}(v_1) + P(v_2|\neg v_1)\pi_{V_2}^{V_1}(\neg v_1) \\ &= 0.4 \times 0.8 + 0.9 \times 0.2 = 0.5 \\ P(\neg v_2) &= P(\neg v_2|v_1)\pi_{V_2}^{V_1}(v_1) + P(\neg v_2|\neg v_1)\pi_{V_2}^{V_1}(\neg v_1) \\ &= 0.6 \times 0.8 + 0.1 \times 0.2 = 0.5 \end{aligned}$$

Message passing: three vertices



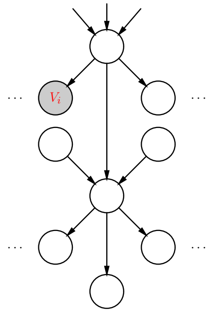
It holds that: $\pi_{V_3}^{V_2}(v_2) = P(v_2)$ and $\pi_{V_3}^{V_2}(\neg v_2) = P(\neg v_2)$

Local computation in V_3:

$$\begin{aligned} P(v_3) &= P(v_3|v_2)\pi_{V_3}^{V_2}(v_2) + P(v_3|\neg v_2)\pi_{V_3}^{V_2}(\neg v_2) \\ P(\neg v_3) &= P(\neg v_3|v_2)\pi_{V_3}^{V_2}(v_2) + P(\neg v_3|\neg v_2)\pi_{V_3}^{V_2}(\neg v_2) \end{aligned}$$

Evidence propagation

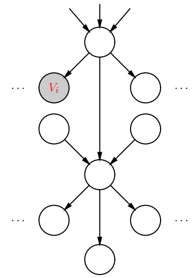
Let $\mathcal{B} = (G, P)$ be a Bayesian network with digraph G and joint probability distribution P



- Evidence is an assignment of value to a variable (i.e., instantiating the variable): $V_i =$ *true* ($= v_i$) or $V_i =$ *false* ($= \neg v_i$) for binary variable V_i

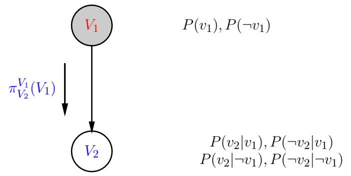
Evidence propagation (cont)

Let $\mathcal{B} = (G, P)$ be a Bayesian network



- Given an evidence, P no longer holds and must be updated to a new probability distribution P^*. E.g., for evidence v_i it holds that $P^*(v_i) = 1$ ($P^*(\neg v_i) = 0$), whereas originally $P(v_i) = 0.3$ ($P(\neg v_i) = 0.7$).
- Entire Bayesian network must be updated

Evidence and causal parameter



Evidence: assume that $V_1 = \textit{true}$ $(= v_1)$

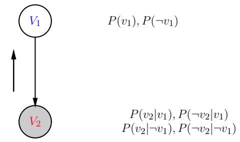
$$\pi_{V_2}^{V_1}(v_1) = 1,\ \pi_{V_2}^{V_1}(\neg v_1) = 0$$

Local computation in V_2:

$$\begin{aligned} P^*(v_2) &= P(v_2|v_1)\pi_{V_2}^{V_1}(v_1) + P(v_2|\neg v_1)\pi_{V_2}^{V_1}(\neg v_1) \\ &= P(v_2|v_1) \end{aligned}$$

$$\begin{aligned} P^*(\neg v_2) &= P(\neg v_2|v_1)\pi_{V_2}^{V_1}(v_1) + P(\neg v_2|\neg v_1)\pi_{V_2}^{V_1}(\neg v_1) \\ &= P(\neg v_2|v_1) \end{aligned}$$

Evidence and diagnostic parameter



Evidence: assume that $V_2 = \textit{true}$ $(= v_2)$

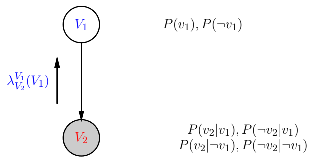
$$P^*(v_2) = 1\ ,\ P^*(\neg v_2) = 0$$

Updated probability distribution $P^*(V_1)$:

$$\begin{aligned} P^*(v_1) &= P(v_1|v_2) = \frac{P(v_2|v_1)P(v_1)}{P(v_2)} \\ P^*(\neg v_1) &= P(\neg v_1|v_2) = \frac{P(v_2|\neg v_1)P(\neg v_1)}{P(v_2)} \end{aligned}$$

for which V_1 needs $P(V_2|V_1)$ from V_2: message $\lambda_{V_2}^{V_1}(V_1)$

Evidence and diagnostic parameter (cont)



Evidence: assume that $V_2 = \textit{true}$ $(= v_2)$

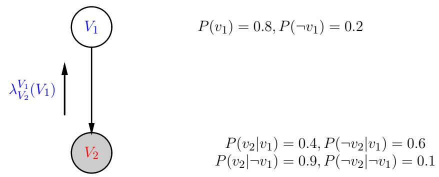
V_2 sends a message $\lambda_{V_2}^{V_1}(V_1)$ to V_1 so V_1 can compute $P^*(V_1)$

This message is defined as follows:

$$\begin{aligned} \lambda_{V_2}^{V_1}(v_1) &= P(v_2|v_1) \\ \lambda_{V_2}^{V_1}(\neg v_1) &= P(v_2|\neg v_1) \end{aligned}$$

$\lambda_{V_2}^{V_1}(V_1)$ is called the diagnostic parameter

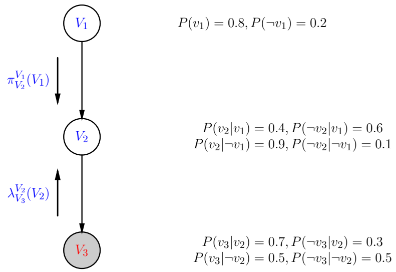
Example: diagnostic parameter



Updated probability distribution $P^*(V_1)$:

$$\begin{aligned} P^*(v_1) &= \frac{P(v_2|v_1)P(v_1)}{P(v_2)} = \alpha\, \lambda_{V_2}^{V_1}(v_1)P(v_1) \\ &= \alpha \times 0.4 \times 0.8 = 0.32\alpha \\ P^*(\neg v_1) &= \frac{P(v_2|\neg v_1)P(\neg v_1)}{P(v_2)} = \alpha\, \lambda_{V_2}^{V_1}(\neg v_1)P(\neg v_1) \\ &= \alpha \times 0.9 \times 0.2 = 0.18\alpha \end{aligned}$$

Causal and diagnostic parameters combined



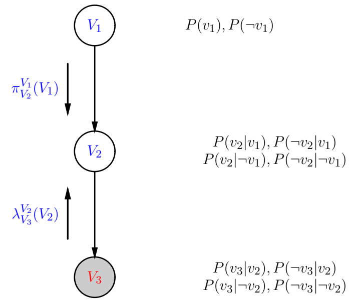
Updated probability distribution $P^*(V_2)$ for *evidence* v_3:

$$P^*(v_2) = \alpha\, \lambda^{V_2}_{V_3}(v_2)[P(v_2|v_1)\pi^{V_1}_{V_2}(v_1) + P(v_2|\neg v_1)\pi^{V_1}_{V_2}(\neg v_1)]$$
$$= \alpha \times 0.7[0.4 \times 0.8 + 0.9 \times 0.2] = 0.35\alpha$$

$P^*(\neg v_2)$ = analogous = 0.25α (thus, $\alpha = 1\frac{2}{3}$)

$\lambda^{V_2}_{V_3}(V_2) = P(V_3|V_2)$, and $\pi^{V_1}_{V_2}(V_1) = P(V_1)$

Towards a generic formula



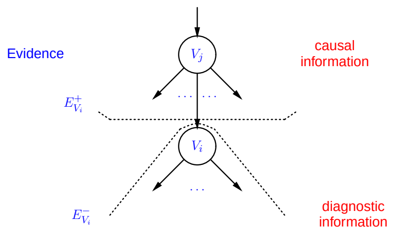
Updated probability distribution $P^*(V_2)$ for *evidence* v_3:

$P^*(V_2)$ = $\alpha\, \lambda^{V_2}_{V_3}(V_2)[P(V_2|v_1)\pi^{V_1}_{V_2}(v_1) + P(V_2|\neg v_1)\pi^{v_1}_{V_2}(\neg v_1)]$

= $\alpha \cdot$ diagnostic information for $V_2 \cdot$ causal information for V_2

= $P(V_2 \mid \text{Evidence})$

Generic formula: data fusion



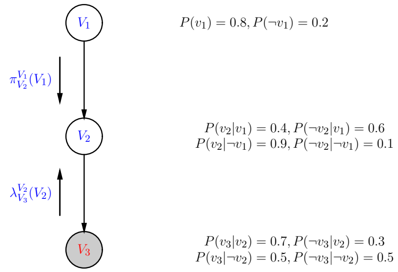
Data fusion:

$P^*(V_i)$ = $P(V_i \mid E)$ = $\alpha \cdot \pi(V_i) \cdot \lambda(V_i)$

where:

- E: evidence
- α: normalisation constant
- $\pi(V_i)$: *compound* causal parameter
- $\lambda(V_i)$: *compound* diagnostic parameter

Example of data fusion

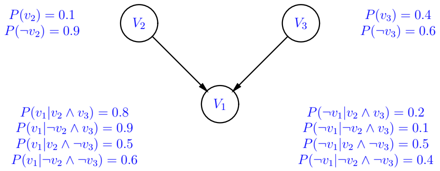


Evidence v_3:

- $\lambda(v_2) = \lambda^{V_2}_{V_3}(v_2) = 0.7$
- $\pi(v_2) = P(v_2|v_1)\pi^{V_1}_{V_2}(v_1) + P(v_2|\neg v_1)\pi^{V_1}_{V_2}(\neg v_1) = 0.5$
- $\alpha = 1\frac{2}{3}$

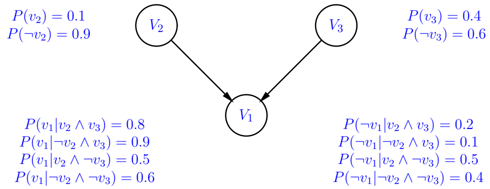
$$P^*(v_2) = P(v_2 \mid v_3) = \alpha \cdot \pi(v_2) \cdot \lambda(v_2) \approx 0.58$$

Simple network example



$$
\begin{aligned}
P(v_1) &= \alpha \cdot \pi(v_1) \cdot \lambda(v_1) \\
P(\neg v_1) &= \alpha \cdot \pi(\neg v_1) \cdot \lambda(\neg v_1) \\
\pi(v_1) &= P(v_1|v_2 \wedge v_3)\pi^{V_2}_{V_1}(v_2)\pi^{V_3}_{V_1}(v_3) + \\
&\quad P(v_1|\neg v_2 \wedge v_3)\pi^{V_2}_{V_1}(\neg v_2)\pi^{V_3}_{V_1}(v_3) + \\
&\quad P(v_1|v_2 \wedge \neg v_3)\pi^{V_2}_{V_1}(v_2)\pi^{V_3}_{V_1}(\neg v_3) + \\
&\quad P(v_1|\neg v_2 \wedge \neg v_3)\pi^{V_2}_{V_1}(\neg v_2)\pi^{V_3}_{V_1}(\neg v_3) \\
&= 0.71 \\
\pi(\neg v_1) &= 0.29
\end{aligned}
$$

Simple network example (cont)

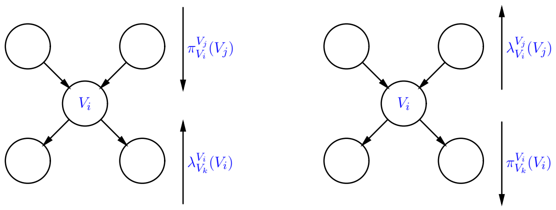


$$
\begin{aligned}
P(v_1) &= \alpha \cdot \pi(v_1) \cdot \lambda(v_1) \\
P(\neg v_1) &= \alpha \cdot \pi(\neg v_1) \cdot \lambda(\neg v_1)
\end{aligned}
$$

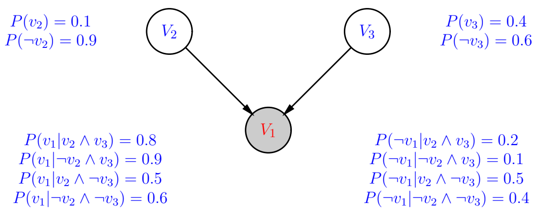
Given that no evidence is provided, $\lambda(v_1) = 1$.
Analogously, $\lambda(\neg v_1) = 1$.
Therefore,

- $P(v_1) = \alpha \cdot 0.71 \cdot 1$; $P(\neg v_1) = \alpha \cdot 0.29 \cdot 1$
- $\Rightarrow \alpha = 1$

Messages to children and parents

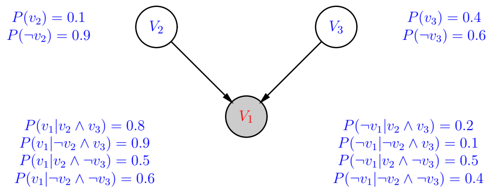


Example:



- Suppose that the evidence $V_1 = true$ is observed, we want to compute the updated probability of V_2

Messages to child and parent (cont)



The probabilities of interest are computed according to the fusion lemma:

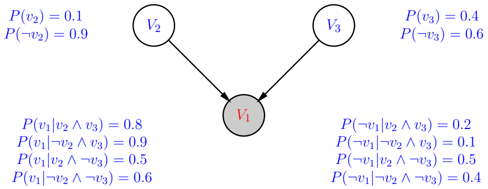
$$P^*(V_2) = \alpha \cdot \pi(V_2) \cdot \lambda(V_2)$$

V_2 has now to compute its compound parameters

Having no children, the compound causal parameter for V_2 is then:

$$
\begin{aligned}
\pi(v_2) &= P(v_2) \\
\pi(\neg v_2) &= P(\neg v_2)
\end{aligned}
$$

Messages to child and parent (cont)



The values of the compound diagnostic parameter are calculated from

$$\lambda(v_2) = \lambda_{V_1}^{V_2}(v_2)$$
$$\lambda(\neg v_2) = \lambda_{V_1}^{V_2}(\neg v_2)$$

From its sucessor V_1, vertex V_2 receives the diagnostic parameter $\lambda_{V_1}^{V_2}$

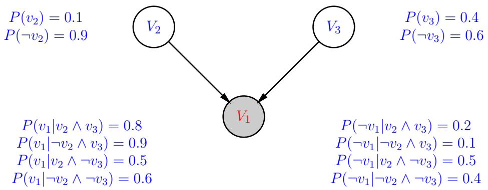
Messages to child and parent (cont)

Such diagnostic parameter has values:

$$\begin{aligned}
\lambda_{V_1}^{V_2}(v_2) &= \lambda(v_1) \cdot [P(v_1|v_2 \wedge v_3)\pi_{V_1}^{V_3}(v_3) + \\
&\qquad P(v_1|v_2 \wedge \neg v_3)\pi_{V_1}^{V_3}(\neg v_3)] + \\
&\quad \lambda(\neg v_1) \cdot [P(\neg v_1|v_2 \wedge v_3)\pi_{V_1}^{V_3}(v_3) + \\
&\qquad P(\neg v_1|v_2 \wedge \neg v_3)\pi_{V_1}^{V_3}(\neg v_3)] \\
&= \lambda(v_1) \cdot [P(v_1|v_2 \wedge v_3)\pi_{V_1}^{V_3}(v_3) + \\
&\qquad P(v_1|v_2 \wedge \neg v_3)\pi_{V_1}^{V_3}(\neg v_3)] \\
&= 0.8 \times 0.4 + 0.5 \times 0.6 \\
&= 0.62
\end{aligned}$$

Analogously for $\lambda_{V_1}^{V_2}(\neg v_2) = 0.72$

Messages to child and parent (cont)



$$P^*(V_2) = \alpha \cdot \pi(V_2) \cdot \lambda(V_2)$$
$$\lambda_{V_1}^{V_2}(v_2) = 0.62 \text{ and } \lambda_{V_1}^{V_2}(\neg v_2) = 0.72$$
$$\pi(v_2) = 0.1 \text{ and } \pi(\neg v_2) = 0.9$$

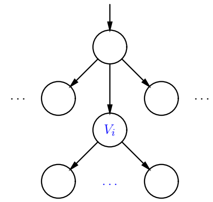
Result:

$$\begin{aligned}
P^*(v_2) &= \alpha \cdot 0.1 \cdot 0.62 = 0.062\alpha \\
P^*(v_2) &= \alpha \cdot 0.9 \cdot 0.72 = 0.648\alpha
\end{aligned}$$

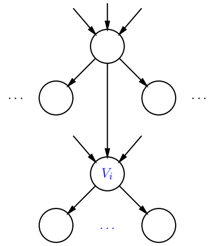
$\Rightarrow P^*(v_2) \approx 0.087, P^*(\neg v_2) \approx 0.913$

Topology of Bayesian networks

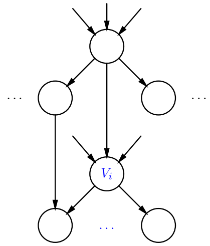
- Directed tree:



- Singly/multiply-connected network:

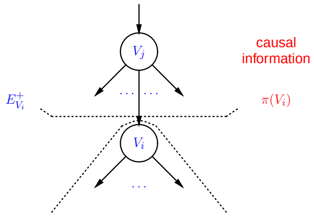


singly-connected



multiply-connected

Compound causal parameters for trees



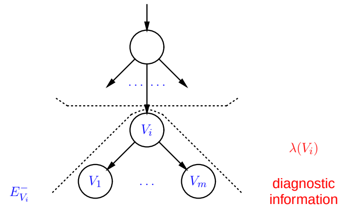
Bayesian network $\mathcal{B} = (G, P)$, where

- $G = (V(G), A(G))$ is a directed tree, i.e. exactly one directed path from the root to each vertex $V_i \in V(G)$
- P is a joint probability distribution on $V(G)$

$$\pi(V_i) = \sum_{V_j} P(V_i \mid V_j) \cdot \pi_{V_i}^{V_j}(V_j)$$

with evidence $E^+_{V_i}$ incorporated into $\pi_{V_i}^{V_j}(V_j)$

Compound diagnostic parameters for trees



Bayesian network $\mathcal{B} = (G, P)$, where

- $G = (V(G), A(G))$ is a directed tree
- P is the joint probability distribution on $V(G)$

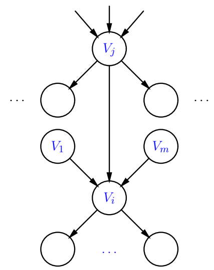
It holds that:

$$\lambda(V_i) = \prod_{j=1}^{m} \lambda_{V_j}^{V_i}(V_i)$$

with evidence $E^-_{V_i}$ incorporated into $\lambda_{V_j}^{V_i}(V_i)$

Compound causal parameter for SCN

- Data fusion: $P^*(V_i) = P(V_i \mid E) = \alpha \cdot \pi(V_i) \cdot \lambda(V_i)$
- Compound causal parameter $\pi(V_i)$:

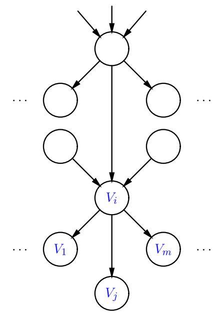


$$\pi(V_i) = \sum_{\rho(V_i)} P(V_i \mid \rho(V_i)) \cdot \prod_{j=1}^{m} \pi_{V_i}^{V_j}(V_j)$$

with parents $\rho(V_i) = V_1 \wedge \cdots \wedge V_j \wedge \cdots \wedge V_m$

Compound diagnostic parameter for SCN

- Data fusion: $P^*(V_i) = P(V_i \mid E) = \alpha \cdot \pi(V_i) \cdot \lambda(V_i)$
- Compound diagnostic parameter $\lambda(V_i)$:

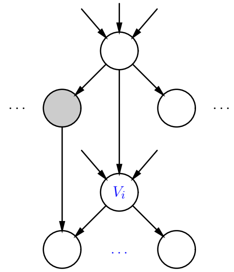


$$\lambda(V_i) = \prod_{j=1}^{m} \lambda_{V_j}^{V_i}(V_i)$$

Overview of Pearl's algorithm

- All the computations are local
- Efficient for local computation property and parallel, distributed implementations
- However, there is a summation over all joint instantiations of parent nodes - exponential in the number of parents
  - if parents sets are bound in size by a constant, the runtime is *linear*
- Therefore, computationally infeasible in networks where nodes have too many parents
- Number of data propagation cycles proportional to the length of path(s) from evidence node(s)

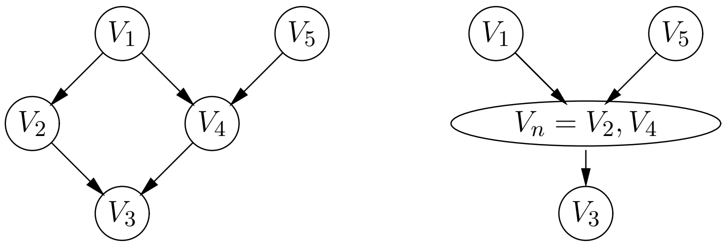
Multiply-connected networks inference



- At least two nodes are connected by more than one path (in the underlying undirected path)
- Thus, some variables can influence another through more than one causal mechanism
- And same evidence counted more than once

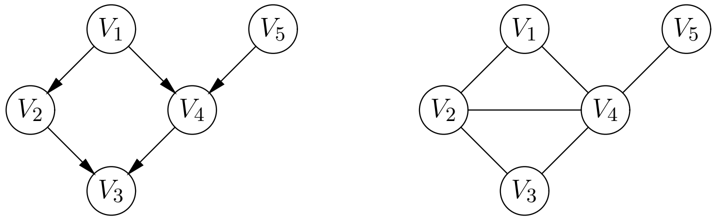
Clustering inference algorithm

- Transform a BN into a equivalent polytree by merging nodes
  - Removal of multiple paths between nodes
  - New node has as states all possible instantiations of combined nodes
- Probabilities updating on transformed polytree

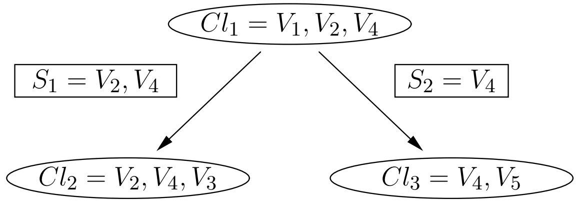


(a) (b)

Example



(a) (b)



(c)

References

- Kim, J. H. and Pearl, J. (1983) A computational model for causal and diagnostic reasoning in inference systems. In *Proceeding of the Eighth International Joint Conference in Artificial Intelligence (IJCAI)*, pp. 190-193.
- Pearl, J. (1988) *Probabilistic Reasoning in Intelligent Systems*. Morgan Kauffman. ISBN 0-934613-73-7.
- Spiegelhalter, D. J. and Lauritzen, S. L. (1990) Sequential updating of conditional probabilities on directed graphical structures. *Networks*, **20**, pp. 579-605.
- Jensen, F. V. and Nielsen, T. D. *Bayesian Networks and Decision Graphs*. Springer. ISBN 978-0-387-68281