Syllabus, Chapter 4:

The Bayesian Network Framework

The network formalism, informal

A Bayesian network combines two types of domain knowledge to represent a joint probability distribution:

- qualitative knowledge: a (minimal) directed I-map for the independence relation that exists on the variables of the domain;
- quantitative knowledge: a set of local conditional probability distributions.

A Bayesian network

Definition:

A Bayesian network is a pair $\mathcal{B} = (G, \Gamma)$ such that

- $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$ is a DAG with arcs $\boldsymbol{A}_G$ and nodes $\boldsymbol{V}_G = \boldsymbol{V}$, representing a set of random variables $\boldsymbol{V} = \{V_1, \ldots, V_n\}$, $n \geq 1$;
- $\Gamma = \{\gamma_{V_i} \mid V_i \in \boldsymbol{V}\}$ is a set of non-negative functions

$$\gamma_{V_i} : \{c_{V_i}\} \times \{c_{\boldsymbol{\rho}(V_i)}\} \rightarrow [0, 1]$$

such that for each configuration $c_{\boldsymbol{\rho}(V_i)}$ of the set $\boldsymbol{\rho}(V_i)$ of parents of V_i in G, we have that

$$\sum_{c_{V_i}} \gamma_{V_i}(c_{V_i} \mid c_{\boldsymbol{\rho}(V_i)}) = 1 \quad \text{for } i = 1, \ldots n$$

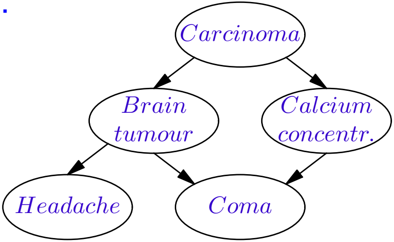
These functions are called the assessment functions for G; their values are referred to as network- or model-parameters.

An Example

Consider the following piece of 'medical knowledge':

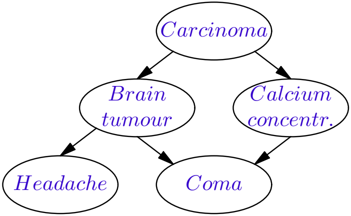
> *"A metastatic carcinoma can cause a brain tumour and is also a possible explanation for an increased concentration of calcium in the blood. Both a brain tumour and an increased calcium concentration can result in a patient falling into a coma. A brain tumour can cause severe headaches."*

The independences between the variables are represented in the following DAG G:



An example – continued

Reconsider the following DAG G, and assume each $V \in \boldsymbol{V}$ to be binary-valued.



With G we associate a set of assessment functions $\Gamma = \{\gamma_{Car}, \gamma_B, \gamma_{Cal}, \gamma_H, \gamma_{Co}\}$.

For the function γ_{Car} the following function values are specified:

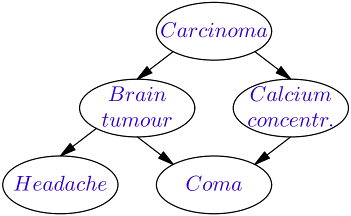
$$\gamma_{Car}(carc) = 0.2, \quad \gamma_{Car}(\neg\, carc) = 0.8$$

For the function γ_B the following function values are specified:

$$\gamma_B(tum \mid carc) = 0.2, \qquad \gamma_B(tum \mid \neg\, carc) = 0.05$$
$$\gamma_B(\neg\, tum \mid carc) = 0.8, \quad \gamma_B(\neg\, tum \mid \neg\, carc) = 0.95$$

An example – continued

Reconsider the following DAG G, and assume each $V \in \boldsymbol{V}$ to be binary-valued.



With G we associate a set of assessment functions $\Gamma = \{\gamma_{Car}, \gamma_B, \gamma_{Cal}, \gamma_H, \gamma_{Co}\}$.

For the function γ_{Cal} the following function values are specified:

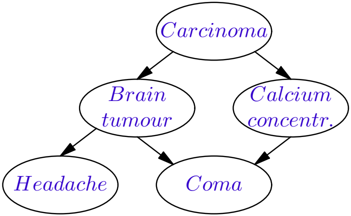
$$\gamma_{Cal}(cal\ conc \mid carc) = 0.8 \qquad \gamma_{Cal}(cal\ conc \mid \neg\, carc) = 0.1$$
$$\gamma_{Cal}(\neg cal\ conc \mid carc) = 0.2 \quad \gamma_{Cal}(\neg cal\ conc \mid \neg carc) = 0.9$$

For the function γ_H the following function values are specified:

$$\gamma_H(headache \mid tum) = 0.8 \qquad \gamma_H(headache \mid \neg\, tum) = 0.6$$
$$\gamma_H(\neg\, headache \mid tum) = 0.2 \quad \gamma_H(\neg\, headache \mid \neg\, tum) = 0.4$$

An example – continued

Reconsider the following DAG G, and assume each $V \in \boldsymbol{V}$ to be binary-valued.



With G we associate a set of assessment functions $\Gamma = \{\gamma_{Car}, \gamma_B, \gamma_{Cal}, \gamma_H, \gamma_{Co}\}$.

For the function γ_{Co} the following function values are specified:

$$\gamma_{Co}(co \mid tum \wedge cal\ conc) = 0.9 \qquad \gamma_{Co}(co \mid \neg\, tum \wedge cal\ conc) = 0.8$$
$$\gamma_{Co}(co \mid tum \wedge \neg\, cal\ conc) = 0.7 \qquad \gamma_{Co}(co \mid \neg\, tum \wedge \neg\, cal\ conc) = 0.05$$
$$\gamma_{Co}(\neg co \mid tum \wedge cal\ conc) = 0.1 \qquad \gamma_{Co}(\neg co \mid \neg\, tum \wedge cal\ conc) = 0.2$$
$$\gamma_{Co}(\neg co \mid tum \wedge \neg\, cal\ conc) = 0.3 \qquad \gamma_{Co}(\neg co \mid \neg\, tum \wedge \neg\, cal\ conc) = 0.95$$

The pair $\mathcal{B} = (G, \Gamma)$ is a Bayesian network.

A probabilistic interpretation

Proposition:
Let $\mathcal{B} = (G, \Gamma)$ be a Bayesian network with $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$ and nodes $\boldsymbol{V}_G = \boldsymbol{V}$, representing a set of random variables $\boldsymbol{V} = \{V_1, \ldots, V_n\}$, $n \geq 1$. Then

$$\Pr(\boldsymbol{V}) = \prod_{i=1}^{n} \gamma_{V_i}(V_i \mid \boldsymbol{\rho}(V_i))$$

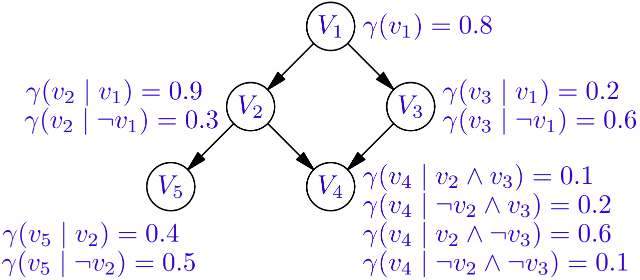
defines a joint probability distribution $\Pr$ on $\boldsymbol{V}$ such that G is a directed I-map for the independence relation $I_{\Pr}$ of $\Pr$.

$\Pr$ is called the joint distribution defined by $\mathcal{B}$ and is said to respect the independences portrayed in G.

NB we will often omit the subscript in γ if no confusion is possible.

An example

Consider the Bayesian network $\mathcal{B}$:



Let $\Pr$ be the joint distribution defined by $\mathcal{B}$. Then, for example

$$
\begin{aligned}
\Pr(v_1 \wedge v_2 \wedge v_3 \wedge v_4 \wedge v_5) &= \\
&= \gamma(v_5 \mid v_2) \cdot \gamma(v_4 \mid v_2 \wedge v_3) \cdot \gamma(v_3 \mid v_1) \cdot \gamma(v_2 \mid v_1) \cdot \gamma(v_1) = \\
&= 0.4 \cdot 0.1 \cdot 0.2 \cdot 0.9 \cdot 0.8 = 0.00576
\end{aligned}
$$

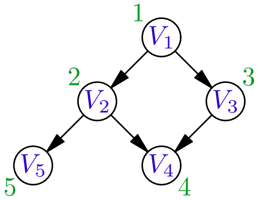
Note: $\Pr$ is described by only 11 (free) model-parameters instead of 31 numbers using a straightforward representation.

A probabilistic interpretation

Proof: (sketch)

Acyclic digraph G allows a total ordering $\iota_G : \boldsymbol{V}_G \leftrightarrow \{1, \ldots, n\}$ such that $\iota_G(V_i) < \iota_G(V_j)$ if there is a *directed path* from V_i to V_j, $i \neq j$, in G.

Example:



A probabilistic interpretation: proof continued

Take ordering ι_G as an ordering on the random variables $V_1, \ldots V_n$ as well.

Let P be an arbitrary joint distribution on $\boldsymbol{V}$ such that G is a directed I-map for the independences in P.

Now apply the chain rule using ι_G.

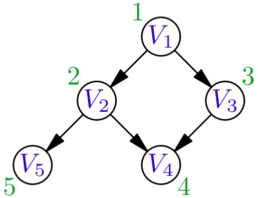
Example:

$$P(V_1 \wedge \ldots \wedge V_5) =$$

$$P(V_5 \mid V_1 \wedge \ldots \wedge V_4) \cdot P(V_4 \mid V_1 \wedge V_2 \wedge V_3) \cdot$$
$$\cdot P(V_3 \mid V_1 \wedge V_2) \cdot P(V_2 \mid V_1) \cdot P(V_1)$$

A probabilistic interpretation: proof continued

Example:



$$P(V_1 \wedge \ldots \wedge V_5) = P(V_5 \mid V_1 \wedge \ldots \wedge V_4) \cdot P(V_4 \mid V_1 \wedge V_2 \wedge V_3) \cdot \cdot P(V_3 \mid V_1 \wedge V_2) \cdot P(V_2 \mid V_1) \cdot P(V_1)$$

Each V_j is conditioned on just those V_i with $\iota_G(V_i) < \iota_G(V_j)$.
Use the fact that G is an I-map for P.

Example: $P(V_1 \wedge \ldots \wedge V_5) = P(V_5 \mid V_2) \cdot P(V_4 \mid V_2 \wedge V_3) \cdot \cdot P(V_3 \mid V_1) \cdot P(V_2 \mid V_1) \cdot P(V_1)$

We have that $P(V_1 \wedge \ldots \wedge V_n) = \prod_{V_i \in \boldsymbol{V}} P(V_i \mid \boldsymbol{\rho}(V_i))$

A probabilistic interpretation: proof continued

With graph G is associated a set Γ of assessment functions $\gamma(V_i \mid \boldsymbol{\rho}(V_i))$. If we choose $\Pr(V_i \mid \boldsymbol{\rho}(V_i)) = \gamma(V_i \mid \boldsymbol{\rho}(V_i))$, then

$$\Pr(V_1 \wedge \ldots \wedge V_n) = \prod_{V_i \in \boldsymbol{V}} \gamma(V_i \mid \boldsymbol{\rho}(V_i))$$

defines a unique joint distribution on $\boldsymbol{V}$ that respects the independences in G.

Example: The joint distribution $\Pr$ defined by

$$\begin{aligned}\Pr(V_1 \wedge \ldots \wedge V_5) = \; & \gamma(V_5 \mid V_2) \cdot \gamma(V_4 \mid V_2 \wedge V_3) \cdot \\ & \cdot \gamma(V_3 \mid V_1) \cdot \gamma(V_2 \mid V_1) \cdot \gamma(V_1)\end{aligned}$$

respects the independences in G.

■

Consequences of probabilistic interpretation

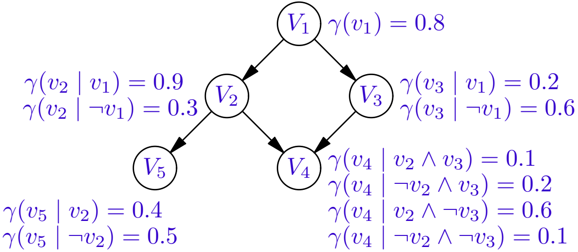
- Bayesian network $\mathcal{B}$ is a very compact representation of a multivariate joint distribution $\Pr(\boldsymbol{V})$, from which we can compute:
  - any prior or marginal probability $\Pr(c_{\boldsymbol{W}})$ for $\boldsymbol{W} \subseteq \boldsymbol{V}$;
  - any posterior or conditional probability $\Pr(c_{\boldsymbol{W}} \mid c_{\boldsymbol{E}})$ for $\boldsymbol{W}, \boldsymbol{E} \subset \boldsymbol{V}$;
- the independences stated in $I_{\Pr}$ are respected by $\mathcal{B}$ and read from graph G by means of the d-separation criterion

  $\Rightarrow$ blocking sets $\boldsymbol{Z}$ now have an intuitive meaning:

  take $\boldsymbol{Z} = \boldsymbol{E}$ upon observing evidence for $\boldsymbol{E} \subset \boldsymbol{V}$.

An example

Let $\mathcal{B} = (G, \Gamma)$ and $\Pr$ be as before.



How can we compute $\Pr(v_1 \wedge v_3 \wedge v_4 \wedge v_5)$?

$$\Pr(v_1 \wedge v_2 \wedge v_3 \wedge v_4 \wedge v_5) = 0.00576$$

$$\Pr(v_1 \wedge \neg\, v_2 \wedge v_3 \wedge v_4 \wedge v_5) = 0.0016$$

$$\begin{aligned}\Pr(v_1 \wedge v_3 \wedge v_4 \wedge v_5) &= \\ &= \Pr(v_1 \wedge v_2 \wedge v_3 \wedge v_4 \wedge v_5) + \Pr(v_1 \wedge \neg v_2 \wedge v_3 \wedge v_4 \wedge v_5) \\ &= 0.00576 + 0.0016 = 0.00736\end{aligned}$$

Exact inference algorithms

Efficiently compute marginal and conditional probabilities from the distribution defined by a network.

The best-known algorithms serve to compute univariate distributions over $V_i \in \boldsymbol{V}$, i.e. $\Pr(V_i)$ or $\Pr(V_i \mid c_{\boldsymbol{E}})$:

- Belief propagation (BP) (J. Pearl (1986). *Fusion, propagation and structuring in belief networks,* Artificial Intelligence, 29);
- Join-tree propagation (S.L. Lauritzen, D.J. Spiegelhalter (1988). *Local computations with probabilities on graphical structures and their application to expert systems,* Journal of the Royal Statistical Society (Series B), 50);
- Variable elimination (N.L. Zhang, D. Poole (1994). *A simple approach to Bayesian network computations,* 7th Canadian Conference on AI).

The algorithms are quite different in terms of the underlying ideas and their complexity.

Approximate inference algorithms

Estimate probabilities from the distribution defined by a network.

- Loopy belief propagation
- Sampling-based approaches
  - Monte Carlo techniques, e.g. MCMC
  - accurate with enough samples
  - sampling can be computationally demanding
- Deterministic approaches
  - e.g. variational approaches, such as VI
  - use analytical approximations to the posterior
  - can scale well

Variable elimination (VE): idea and complexity

Let $\boldsymbol{V} = \{V_1, V_2, V_3, V_4\}$. Consider the computation of

$$\Pr(v_4) = \sum_{c_{\{V_1,V_2,V_3\}}} \Pr(c_{V_1}) \cdot \Pr(c_{V_2} \mid c_{V_3}) \cdot \Pr(c_{V_3} \mid c_{V_1}) \cdot \Pr(v_4 \mid c_{V_3})$$

- avoid computing large factors: move summations inside the factorisation;
- efficiency depends on size (w(idth)) of largest computed factor, which depends on order of elimination:

$$\sum_{c_{V_1}} \Pr(c_{V_1}) \cdot \sum_{c_{V_3}} \Pr(c_{V_3} \mid c_{V_1}) \cdot \Pr(v_4 \mid c_{V_3}) \cdot \sum_{c_{V_2}} \Pr(c_{V_2} \mid c_{V_3})$$

Complexity for individual $\Pr(V_i \mid c_{\boldsymbol{E}})$: $O(|\boldsymbol{V}| \cdot \exp(w))$

- singly connected graphs: $w = k$ for $k = \max_{V_i} |\boldsymbol{\rho}_G(V_i)|$
- multiply connected graphs: $w \geq k$ can be as large as $|\boldsymbol{V}|$.

Join-tree propagation: idea and complexity

Idea of Join-tree propagation:

1) moralise and *triangulate* G;
2) identify cliques and organise these into a *join tree*;
3) translate Γ into clique potentials;
4) update clique potentials by message passing between cliques in the tree.

Efficiency depends on size of largest clique ($\rightarrow$ width w).

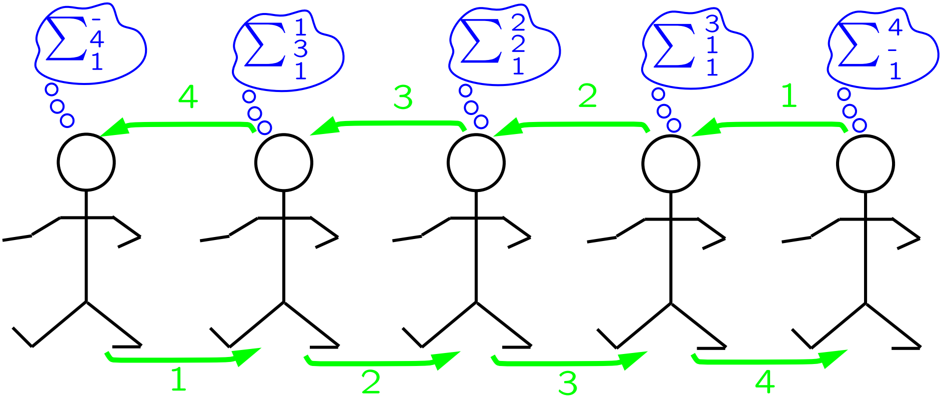
Complexity for all $\Pr(V_i \mid c_{\boldsymbol{E}})$ simultaneously: $O(|\boldsymbol{V}| \cdot \exp(w))$

Pearl’s computational architecture

In *Pearl*’s algorithm the graph of a Bayesian network is used as a computational architecture:

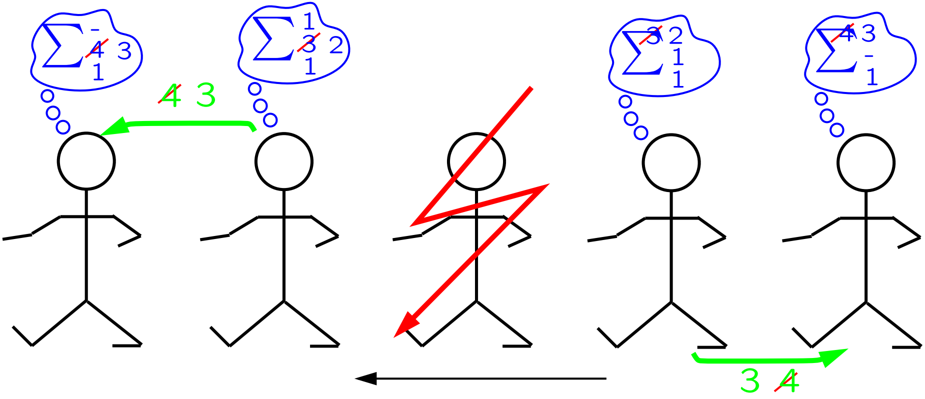
- each node in the graph is an autonomous object;
- each object has a local memory that stores the assessment functions of the associated node;
- each object has available a local processor that can do (simple) probabilistic computations;
- each arc in the graph is a (bi-directional) communication channel, through which connected objects can send each other messages.

A computational architecture



Message-passing and simple local computations:
now we all know with how many we are!

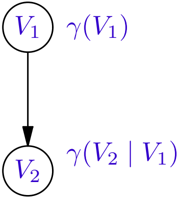
A computational architecture



If we observe a local change:
start message-passing to update computations.

Understanding Pearl: single arc (1)

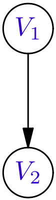
Consider Bayesian network $\mathcal{B}$ with the following graph:



Let $\Pr$ be the joint distribution defined by $\mathcal{B}$.
We consider the situation without evidence.

- What does V_1 need to compute the probabilities $\Pr(V_1)$?
- What does V_2 need to compute the probabilities $\Pr(V_2)$?

Understanding Pearl: single arc (2)



$\gamma(v_1),\ \gamma(\neg v_1)$

$\gamma(v_2 \mid v_1),\ \gamma(\neg v_2 \mid v_1)$
$\gamma(v_2 \mid \neg v_1),\ \gamma(\neg v_2 \mid \neg v_1)$

Let $\Pr$ be the joint distribution defined by $\mathcal{B}$.

We consider the situation without evidence.

- node V_1 can determine the probabilities for its own values:

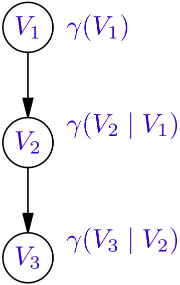
$$\Pr(v_1) = \gamma(v_1), \quad \Pr(\neg v_1) = \gamma(\neg v_1)$$

- node V_2 cannot determine $\Pr(V_2)$, but does know all *four* conditional probabilities: $\Pr(V_2 \mid V_1) = \gamma(V_2 \mid V_1)$

  V_2 can compute its probabilities given information from V_1:

$$\begin{aligned}\Pr(v_2) &= \Pr(v_2 \mid v_1) \cdot \Pr(v_1) + \Pr(v_2 \mid \neg v_1) \cdot \Pr(\neg v_1) \\ \Pr(\neg v_2) &= \Pr(\neg v_2 \mid v_1) \cdot \Pr(v_1) + \Pr(\neg v_2 \mid \neg v_1) \cdot \Pr(\neg v_1)\end{aligned}$$

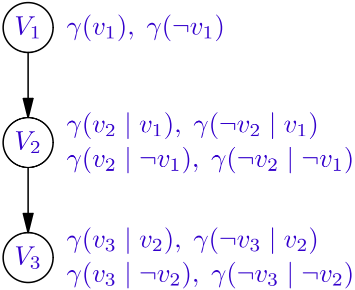
Understanding Pearl: directed path (1)



We consider the situation without evidence.

- What does V_1 need to compute the probabilities $\Pr(V_1)$?
- What does V_2 need to compute the probabilities $\Pr(V_2)$?
- What does V_3 need to compute the probabilities $\Pr(V_3)$?

Understanding Pearl: directed path (2)



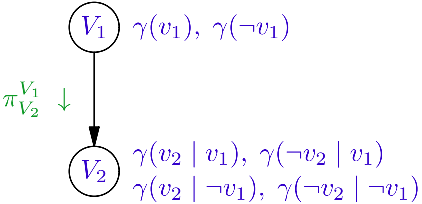
We consider the situation without evidence.

Given information from V_1, node V_2 can compute $\Pr(v_2)$ and $\Pr(\neg v_2)$.

Node V_2 now sends node V_3 the required information; node V_3 computes:

$$\begin{aligned}
\Pr(v_3) &= \Pr(v_3 \mid v_2) \cdot \Pr(v_2) + \Pr(v_3 \mid \neg v_2) \cdot \Pr(\neg v_2) \\
&= \gamma(v_3 \mid v_2) \cdot \Pr(v_2) + \gamma(v_3 \mid \neg v_2) \cdot \Pr(\neg v_2) \\
\Pr(\neg v_3) &= \gamma(\neg v_3 \mid v_2) \cdot \Pr(v_2) + \gamma(\neg v_3 \mid \neg v_2) \cdot \Pr(\neg v_2)
\end{aligned}$$

Introduction to causal message parameters



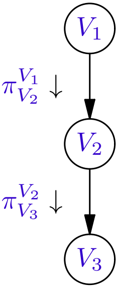
Node V_1 sends a message enabling V_2 to compute the probabilities for its values.

This message contains a function $\pi^{V_1}_{V_2} : \{v_1, \neg v_1\} \to [0, 1]$ for which

$$\sum_{c_{V_1}} \pi^{V_1}_{V_2}(c_{V_1}) = 1$$

$\pi^{V_1}_{V_2}$ is called the causal (message) parameter from V_1 to V_2.

Causal message parameters: an example



$\gamma(v_1) = 0.7,\ \gamma(\neg v_1) = 0.3$

$\gamma(v_2 \mid v_1) = 0.2,\ \gamma(\neg v_2 \mid v_1) = 0.8$
$\gamma(v_2 \mid \neg v_1) = 0.5,\ \gamma(\neg v_2 \mid \neg v_1) = 0.5$

$\gamma(v_3 \mid v_2) = 0.6,\ \gamma(\neg v_3 \mid v_2) = 0.4$
$\gamma(v_3 \mid \neg v_2) = 0.1,\ \gamma(\neg v_3 \mid \neg v_2) = 0.9$

Node V_1:

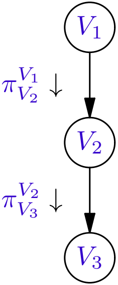
- receives no messages
- computes and sends to V_2: $\pi^{V_1}_{V_2}$

with $\pi^{V_1}_{V_2}(v_1) = \gamma(v_1) = 0.7; \quad \pi^{V_1}_{V_2}(\neg v_1) = 0.3$

Node V_1 computes $\Pr(V_1)$:

$$\Pr(v_1) = \pi^{V_1}_{V_2}(v_1) = 0.7; \quad \Pr(\neg v_1) = 0.3$$

Causal message parameters: an example (cntd)



$\gamma(v_1) = 0.7,\ \gamma(\neg v_1) = 0.3$

$\gamma(v_2 \mid v_1) = 0.2,\ \gamma(\neg v_2 \mid v_1) = 0.8$
$\gamma(v_2 \mid \neg v_1) = 0.5,\ \gamma(\neg v_2 \mid \neg v_1) = 0.5$

$\gamma(v_3 \mid v_2) = 0.6,\ \gamma(\neg v_3 \mid v_2) = 0.4$
$\gamma(v_3 \mid \neg v_2) = 0.1,\ \gamma(\neg v_3 \mid \neg v_2) = 0.9$

Node V_2:

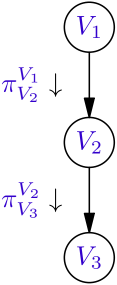
- receives $\pi_{V_2}^{V_1}$ from V_1
- computes and sends to V_3: $\pi_{V_3}^{V_2}$

with
$$\begin{aligned}
\pi_{V_3}^{V_2}(v_2) &= \Pr(v_2 \mid v_1) \cdot \Pr(v_1) + \Pr(v_2 \mid \neg v_1) \cdot \Pr(\neg v_1) \\
&= \gamma(v_2 \mid v_1) \cdot \pi_{V_2}^{V_1}(v_1) + \gamma(v_2 \mid \neg v_1) \cdot \pi_{V_2}^{V_1}(\neg v_1) \\
&= 0.2 \cdot 0.7 + 0.5 \cdot 0.3 = 0.29 \\
\pi_{V_3}^{V_2}(\neg v_2) &= 0.8 \cdot 0.7 + 0.5 \cdot 0.3 = 0.71
\end{aligned}$$

Node V_2 computes $\Pr(V_2)$:

$$\Pr(v_2) = \pi_{V_3}^{V_2}(v_2) = 0.29; \quad \Pr(\neg v_2) = 0.71$$

Causal message parameters: an example (cntd)



$\gamma(v_1) = 0.7,\ \gamma(\neg v_1) = 0.3$

$\gamma(v_2 \mid v_1) = 0.2,\ \gamma(\neg v_2 \mid v_1) = 0.8$
$\gamma(v_2 \mid \neg v_1) = 0.5,\ \gamma(\neg v_2 \mid \neg v_1) = 0.5$

$\gamma(v_3 \mid v_2) = 0.6,\ \gamma(\neg v_3 \mid v_2) = 0.4$
$\gamma(v_3 \mid \neg v_2) = 0.1,\ \gamma(\neg v_3 \mid \neg v_2) = 0.9$

Node V_3:

- receives $\pi^{V_2}_{V_3}$ from V_2
- sends no messages

Node V_3 computes $\Pr(V_3)$:

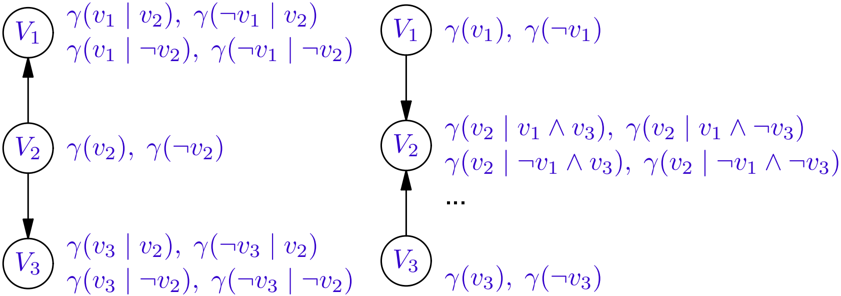
$$\begin{aligned}\Pr(v_3) &= \gamma(v_3 \mid v_2) \cdot \pi^{V_2}_{V_3}(v_2) + \gamma(v_3 \mid \neg v_2) \cdot \pi^{V_2}_{V_3}(\neg v_2) \\ &= 0.6 \cdot 0.29 + 0.1 \cdot 0.71 = 0.245\end{aligned}$$

$$\Pr(\neg v_3) = 0.4 \cdot 0.29 + 0.9 \cdot 0.71 = 0.755$$

■

Understanding Pearl: simple chains

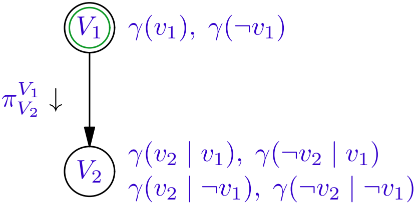
We consider the following networks without observations.



For each network: what information would V_i, $i = 1, 2, 3$, need to compute $\Pr(V_i)$? (consider d-separation and independence)

Understanding Pearl with evidence (1)

Consider $\mathcal{B} = (G, \Gamma)$ with evidence $V_1 = true$ (v_1):



Node V_1 updates its probabilities and causal parameter:

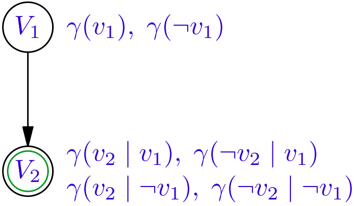
$$\begin{aligned} \pi_{V_2}^{V_1}(v_1) &= \Pr^{v_1}(v_1) \\ &= \Pr(v_1 \mid v_1) = 1 \\ \pi_{V_2}^{V_1}(\neg v_1) &= \Pr^{v_1}(\neg v_1) = 0 \end{aligned}$$

Given the updated information from V_1, node V_2 updates the probabilities for its own values:

$$\begin{aligned} \Pr^{v_1}(v_2) &= \gamma(v_2 \mid v_1) \cdot \pi_{V_2}^{V_1}(v_1) + \gamma(v_2 \mid \neg v_1) \cdot \pi_{V_2}^{V_1}(\neg v_1) \\ &= \gamma(v_2 \mid v_1) \\ \Pr^{v_1}(\neg v_2) &= \gamma(\neg v_2 \mid v_1) \cdot \pi_{V_2}^{V_1}(v_1) + \gamma(\neg v_2 \mid \neg v_1) \cdot \pi_{V_2}^{V_1}(\neg v_1) \\ &= \gamma(\neg v_2 \mid v_1) \end{aligned}$$

Note that the function $\gamma(V_1)$ remains unchanged!

Understanding Pearl with evidence (2a)

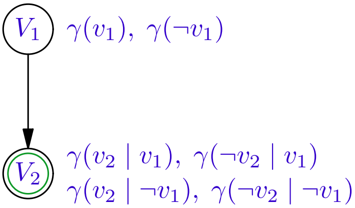


Suppose we have evidence $V_2 = true$ for node V_2.

- What does V_1 need to compute the probabilities $\Pr^{v_2}(V_1)$?
- What does V_2 need to compute the probabilities $\Pr^{v_2}(V_2)$?

Understanding Pearl with evidence (2b)

Consider $\mathcal{B} = (G, \Gamma)$ with evidence $V_2 = true$:



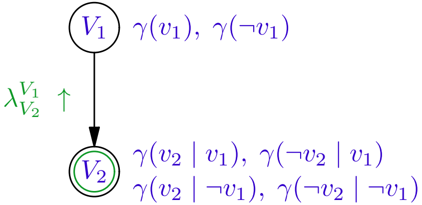
Node V_1 cannot update its probabilities using its own knowledge; it requires information from V_2! What information does V_1 require?

Consider the following properties:

$$\Pr^{v_2}(v_1) = \frac{\Pr(v_2 \mid v_1) \cdot \Pr(v_1)}{\Pr(v_2)} \propto \Pr(v_2 \mid v_1) \cdot \Pr(v_1)$$

$$\Pr^{v_2}(\neg v_1) = \frac{\Pr(v_2 \mid \neg v_1) \cdot \Pr(\neg v_1)}{\Pr(v_2)} \propto \Pr(v_2 \mid \neg v_1) \cdot \Pr(\neg v_1)$$

Introduction to diagnostic message parameters



Node V_2 sends a message enabling V_1 to update the probabilities for its values.

This message contains a function $\lambda^{V_1}_{V_2} : \{v_1, \neg v_1\} \to [0, 1]$ defined on each value of V_1.

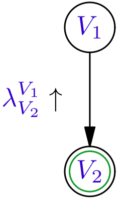
The message basically tells V_1 what node V_2 knows about V_1; in general:

$$\sum_{c_{V_1}} \lambda^{V_1}_{V_2}(c_{V_1}) \neq 1$$

$\lambda^{V_1}_{V_2}$ is called the diagnostic (message) parameter from V_2 to V_1.

Diagnostic message parameters: an example

Consider $\mathcal{B} = (G, \Gamma)$ with evidence $V_2 = true$:



$\gamma(v_1) = 0.8,\ \gamma(\neg v_1) = 0.2$

$\gamma(v_2 \mid v_1) = 0.4,\ \gamma(\neg v_2 \mid v_1) = 0.6$
$\gamma(v_2 \mid \neg v_1) = 0.9,\ \gamma(\neg v_2 \mid \neg v_1) = 0.1$

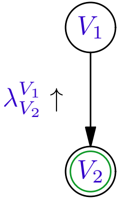
Node V_2:

- computes and sends to V_1: diagnostic parameter $\lambda^{V_1}_{V_2}$ with

$$\lambda^{V_1}_{V_2}(v_1) = \Pr(v_2 \mid v_1) = \gamma(v_2 \mid v_1) = 0.4$$
$$\lambda^{V_1}_{V_2}(\neg v_1) = \gamma(v_2 \mid \neg v_1) = 0.9$$

Note that $\sum_{c_{V_1}} \lambda(c_{V_1}) = 1.3 > 1$!

Diagnostic message parameters: an example (cntd)



$\gamma(v_1) = 0.8,\ \gamma(\neg v_1) = 0.2$

$\gamma(v_2 \mid v_1) = 0.4,\ \gamma(\neg v_2 \mid v_1) = 0.6$
$\gamma(v_2 \mid \neg v_1) = 0.9,\ \gamma(\neg v_2 \mid \neg v_1) = 0.1$

Node V_1 receives from V_2: $\lambda^{V_1}_{V_2}$

Node V_1 computes:

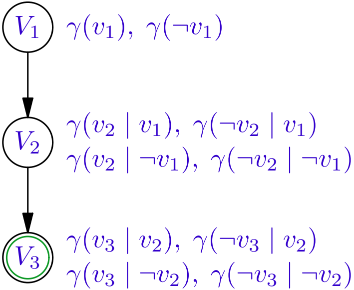
$$\begin{aligned}
\Pr^{v_2}(v_1) &= \alpha \cdot \Pr(v_2 \mid v_1) \cdot \Pr(v_1) \\
&= \alpha \cdot \lambda^{V_1}_{V_2}(v_1) \cdot \gamma(v_1) = \alpha \cdot 0.4 \cdot 0.8 = \alpha \cdot 0.32 \\
\Pr^{v_2}(\neg v_1) &= \alpha \cdot \lambda^{V_1}_{V_2}(\neg v_1) \cdot \gamma(\neg v_1) = \alpha \cdot 0.9 \cdot 0.2 = \alpha \cdot 0.18
\end{aligned}$$

Node V_1 now normalises its probabilities using

$$\Pr^{v_2}(v_1) + \Pr^{v_2}(\neg v_1) = 1 : \alpha \cdot 0.32 + \alpha \cdot 0.18 = 1 \quad \Longrightarrow \alpha = 2$$

resulting in $\Pr^{v_2}(v_1) = 0.64 \quad \Pr^{v_2}(\neg v_1) = 0.36$ ■

Understanding Pearl: directed path with evidence



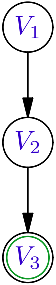
Suppose we have evidence $V_3 = true$ for node V_3.

- What does V_1 need to compute the probabilities $\Pr^{v_3}(V_1)$?
- What does V_2 need to compute the probabilities $\Pr^{v_3}(V_2)$?
- What does V_3 need to compute the probabilities $\Pr^{v_3}(V_3)$?

What if node V_1, node V_2, or both have evidence instead?

Pearl on directed paths – An example (1)

Consider $\mathcal{B} = (G, \Gamma)$ with evidence $V_3 = true$:



$\gamma(v_1),\ \gamma(\neg v_1)$

$\gamma(v_2 \mid v_1),\ \gamma(\neg v_2 \mid v_1)$
$\gamma(v_2 \mid \neg v_1),\ \gamma(\neg v_2 \mid \neg v_1)$

$\gamma(v_3 \mid v_2),\ \gamma(\neg v_3 \mid v_2)$
$\gamma(v_3 \mid \neg v_2),\ \gamma(\neg v_3 \mid \neg v_2)$

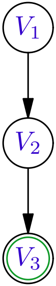
Node V_1:

- receives $\lambda^{V_1}_{V_2}(V_1)$
- computes and sends to V_2: $\pi^{V_1}_{V_2}(V_1) = \gamma(V_1)$

Node V_1 computes

$$\Pr^{v_3}(v_1) = \alpha \cdot \Pr(v_3 \mid v_1) \cdot \Pr(v_1) = \alpha \cdot \lambda^{V_1}_{V_2}(v_1) \cdot \gamma(v_1)$$
$$\Pr^{v_3}(\neg v_1) = \alpha \cdot \Pr(v_3 \mid \neg v_1) \cdot \Pr(\neg v_1) = \alpha \cdot \lambda^{V_1}_{V_2}(\neg v_1) \cdot \gamma(\neg v_1)$$

Pearl on directed paths – An example (2)



$\gamma(v_1),\ \gamma(\neg v_1)$

$\gamma(v_2 \mid v_1),\ \gamma(\neg v_2 \mid v_1)$
$\gamma(v_2 \mid \neg v_1),\ \gamma(\neg v_2 \mid \neg v_1)$

$\gamma(v_3 \mid v_2),\ \gamma(\neg v_3 \mid v_2)$
$\gamma(v_3 \mid \neg v_2),\ \gamma(\neg v_3 \mid \neg v_2)$

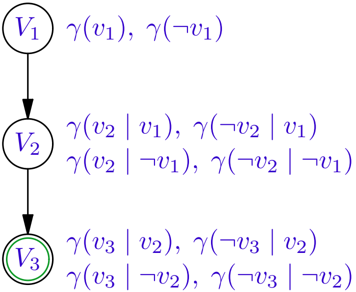
Node V_2:

- receives $\pi^{V_1}_{V_2}(V_1)$ and $\lambda^{V_2}_{V_3}(V_2)$
- computes and sends to V_3: $\pi^{V_2}_{V_3}(V_2)$
- computes and sends to V_1: $\lambda^{V_1}_{V_2}(V_1)$

$$\begin{aligned}
\text{with } \lambda^{V_1}_{V_2}(v_1) &= \Pr(v_3 \mid v_1) \\
&= \Pr(v_3 \mid v_2) \cdot \Pr(v_2 \mid v_1) + \Pr(v_3 \mid \neg v_2) \cdot \Pr(\neg v_2 \mid v_1) \\
&= \lambda^{V_2}_{V_3}(v_2) \cdot \gamma(v_2 \mid v_1) + \lambda^{V_2}_{V_3}(\neg v_2) \cdot \gamma(\neg v_2 \mid v_1) \\
\lambda^{V_1}_{V_2}(\neg v_1) &= \Pr(v_3 \mid \neg v_1) = \ldots
\end{aligned}$$

The node then computes $\Pr^{v_3}(V_2)\ldots$

Pearl on directed paths – An example (3)



Node V_3:

- receives causal parameter $\pi^{V_2}_{V_3}(V_2)$
- computes and sends to V_2: $\lambda^{V_2}_{V_3}(V_2)$ with

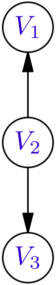
$$\begin{aligned} \lambda^{V_2}_{V_3}(v_2) &= \Pr(v_3 \mid v_2) = \gamma(v_3 \mid v_2) \\ \lambda^{V_2}_{V_3}(\neg v_2) &= \Pr(v_3 \mid \neg v_2) = \gamma(v_3 \mid \neg v_2) \end{aligned}$$

- computes $\Pr^{v_3}(V_3)$

■

Understanding Pearl: simple chain with evidence

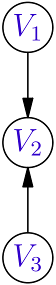
Suppose we have evidence $V_3 = true$ in the following networks:



V_1: $\gamma(v_1 \mid v_2),\ \gamma(\neg v_1 \mid v_2)$
$\gamma(v_1 \mid \neg v_2),\ \gamma(\neg v_1 \mid \neg v_2)$

V_2: $\gamma(v_2),\ \gamma(\neg v_2)$

V_3: $\gamma(v_3 \mid v_2),\ \gamma(\neg v_3 \mid v_2)$
$\gamma(v_3 \mid \neg v_2),\ \gamma(\neg v_3 \mid \neg v_2)$



V_1: $\gamma(v_1),\ \gamma(\neg v_1)$

V_2: $\gamma(v_2 \mid v_1 \wedge v_3),\ \gamma(v_2 \mid v_1 \wedge \neg v_3)$
$\gamma(v_2 \mid \neg v_1 \wedge v_3),\ \gamma(v_2 \mid \neg v_1 \wedge \neg v_3)$
...

V_3: $\gamma(v_3),\ \gamma(\neg v_3)$

For each network: what does node V_i, $i = 1, 2, 3$, need to compute the probabilities $\Pr^{v_3}(V_i)$?

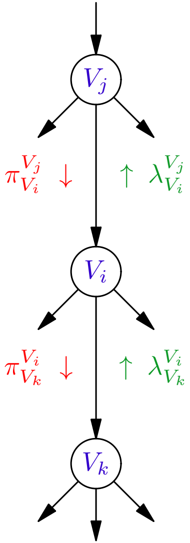
The message parameters

Consider the BN graph as a computational architecture:

causal and diagnostic message parameters

- are passed between objects (nodes)
- through communication channels (arcs).

The causal and diagnostic messages for the same channel are computed independently.



Pearl's algorithm (high-level)

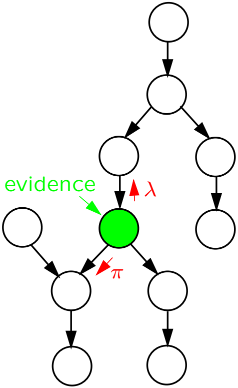
Each $V_i \in \boldsymbol{V}_G$ does the following:

- compute $\pi(V_i)$ once messages from all parents (if any) are received;
- compute $\lambda(V_i)$ once messages from all children (if any) are received;
- for each child V_{i_j}, compute and send message $\pi^{V_i}_{V_{i_j}}(V_i)$ once messages from all other neighbours are received;
- for each parent V_{j_k}, compute and send message $\lambda^{V_{j_k}}_{V_i}(V_{j_k})$ once messages from all other neighbours are received.

Message-passing starts at 'root' and 'leaf' nodes;
upon processing evidence, message-passing is initiated at observed nodes.

The message-passing

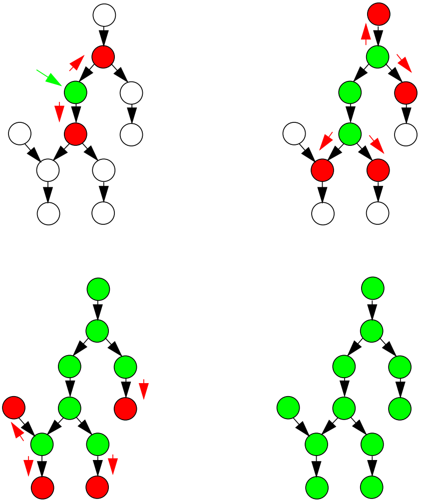
After establishing all prior probabilities, the Bayesian network is in a stable situation.



Once evidence is entered into the network, this stability is disturbed.

The message-passing, continued

Evidence initiates message-passing throughout the network:



After each node in the network is visited by the message-passing algorithm, the network returns to a new stable situation.

Notation: partial configurations

Definition:

A random variable $V_j \in \boldsymbol{V}$ is called instantiated if evidence $V_j = true$ or $V_j = false$ is obtained; otherwise V_j is called uninstantiated.

Let $\boldsymbol{E} \subseteq \boldsymbol{V}$ be *the* subset of instantiated variables. The obtained configuration $c_{\boldsymbol{E}}$ is called a partial configuration of $\boldsymbol{V}$, written $\widetilde{c}_{\boldsymbol{V}}$.

Example: Consider $\boldsymbol{V} = \{V_1, V_2, V_3\}$.
If no evidence is obtained ($\boldsymbol{E} = \emptyset$) then: $\widetilde{c}_{\boldsymbol{V}} = \mathsf{T}(\mathsf{rue})$

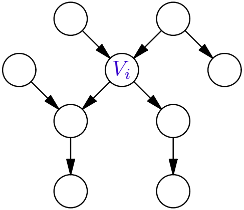
If evidence $V_2 = false$ is obtained, then: $\widetilde{c}_{\boldsymbol{V}} = \neg v_2$ ■

Note: with $\widetilde{c}_{\boldsymbol{V}}$ we can refer to evidence without specifying $\boldsymbol{E}$.

Singly connected graphs (SCGs)

Definition: A directed graph G is called singly connected if the underlying undirected graph of G is acyclic.

Example: The following graph is singly connected:



Lemma: Let G be a singly connected graph (SCG). Each graph obtained from G by removing an arc, is not connected.

Definition: A (directed) tree is a SCG where each node has at most one incoming arc.

Notation: lowergraphs and uppergraphs

Definition: Let $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$ be a SCG and let $G_{(V_i,V_j)}$ be the subgraph of G after removing the arc $(V_i, V_j) \in \boldsymbol{A}_G$:

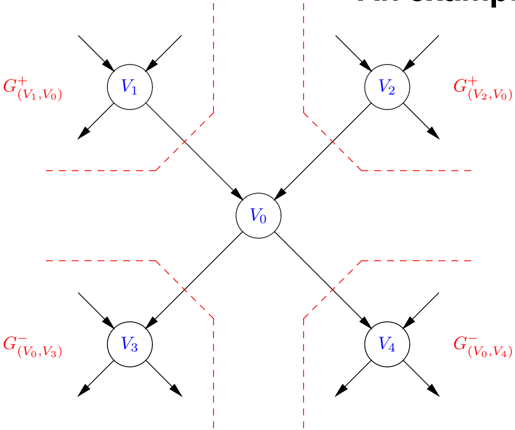
$$G_{(V_i,V_j)} = (\boldsymbol{V}_G, \boldsymbol{A}_G \setminus \{(V_i, V_j)\})$$

Now consider a node $V_i \in \boldsymbol{V}_G$:

For each node $V_j \in \boldsymbol{\rho}(V_i)$, let $G^+_{(V_j,V_i)}$ be the component of $G_{(V_j,V_i)}$ that contains V_j; $G^+_{(V_j,V_i)}$ is called an uppergraph of V_i.

For each node $V_k \in \boldsymbol{\sigma}(V_i)$, let $G^-_{(V_i,V_k)}$ be the component of $G_{(V_i,V_k)}$ that contains V_k; $G^-_{(V_i,V_k)}$ is called a lowergraph of V_i.

An example



Node V_0 has:

– two uppergraphs $G^+_{(V_1,V_0)}$ and $G^+_{(V_2,V_0)}$

– two lowergraphs $G^-_{(V_0,V_3)}$ and $G^-_{(V_0,V_4)}$

For this graph we have, for example, that

$$I(\ \boldsymbol{V}_{G^+_{(V_1,V_0)}},\ \{V_0\},\ \boldsymbol{V}_{G^-_{(V_0,V_3)}}\ )$$

$$I(\ \boldsymbol{V}_{G^-_{(V_0,V_3)}},\ \{V_0\},\ \boldsymbol{V}_{G^-_{(V_0,V_4)}}\ )$$

$$I(\ \boldsymbol{V}_{G^+_{(V_1,V_0)}},\ \emptyset,\ \boldsymbol{V}_{G^+_{(V_2,V_0)}}\ )$$

Computing probabilities in SCGs

Lemma:

Consider $\mathcal{B} = (G, \Gamma)$ with SCG $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$, where $\boldsymbol{V}_G = \boldsymbol{V} = \{V_1, \ldots, V_n\}$, $n \geq 1$; let $\Pr$ be the joint distribution defined by $\mathcal{B}$.

For $V_i \in \boldsymbol{V}$, let $\boldsymbol{V_i^+} = \bigcup_{V_j \in \boldsymbol{\rho}(V_i)} \boldsymbol{V}_{G^+_{(V_j, V_i)}}$ and $\boldsymbol{V_i^-} = \boldsymbol{V} \setminus \boldsymbol{V_i^+}$.

Then

$$\Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}}) = \alpha \cdot \Pr(\widetilde{c}_{\boldsymbol{V_i^-}} \mid V_i) \cdot \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V_i^+}})$$

where $\widetilde{c}_{\boldsymbol{V}} = \widetilde{c}_{\boldsymbol{V_i^-}} \wedge \widetilde{c}_{\boldsymbol{V_i^+}}$ and α is a normalisation constant.

Computing probabilities in SCGs

Proof:

$$\begin{aligned}
\Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}}) &= \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_i^-} \wedge \widetilde{c}_{\boldsymbol{V}_i^+}) \\
&= \frac{\Pr(\widetilde{c}_{\boldsymbol{V}_i^-} \mid V_i) \cdot \Pr(\widetilde{c}_{\boldsymbol{V}_i^+} \mid V_i) \cdot \Pr(V_i)}{\Pr(\widetilde{c}_{\boldsymbol{V}_i^-} \wedge \widetilde{c}_{\boldsymbol{V}_i^+})} \\
&= \Pr(\widetilde{c}_{\boldsymbol{V}_i^-} \mid V_i) \cdot \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_i^+}) \cdot \frac{\Pr(\widetilde{c}_{\boldsymbol{V}_i^+})}{\Pr(\widetilde{c}_{\boldsymbol{V}_i^-} \wedge \widetilde{c}_{\boldsymbol{V}_i^+})} \\
&= \alpha \cdot \Pr(\widetilde{c}_{\boldsymbol{V}_i^-} \mid V_i) \cdot \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_i^+})
\end{aligned}$$

where $\alpha = \dfrac{1}{\Pr(\widetilde{c}_{\boldsymbol{V}_i^-} \mid \widetilde{c}_{\boldsymbol{V}_i^+})}$. ■

Compound parameters: definition

Definition:

Consider $\mathcal{B} = (G, \Gamma)$ with SCG $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$ and joint distribution $\Pr$. For $V_i \in \boldsymbol{V}_G$, let $\boldsymbol{V_i}^+$ and $\boldsymbol{V_i}^-$ be as before;

- the function $\pi : \{v_i, \neg v_i\} \rightarrow [0,1]$ for node V_i is defined by

  $$\pi(V_i) = \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V_i}^+})$$

  and is called the compound causal parameter for V_i;

- the function $\lambda : \{v_i, \neg v_i\} \rightarrow [0,1]$ for node V_i is defined by

  $$\lambda(V_i) = \Pr(\widetilde{c}_{\boldsymbol{V_i}^-} \mid V_i)$$

  and is called the compound diagnostic parameter for V_i.

Computing probabilities in SCGs

Lemma: ('Data Fusion')

Consider $\mathcal{B} = (G, \Gamma)$ with SCG $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$ and joint distribution $\Pr$. Then

$$\text{for each } V_i \in \boldsymbol{V}_G : \qquad \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_G}) = \alpha \cdot \pi(V_i) \cdot \lambda(V_i)$$

with compound causal parameter π, compound diagnostic parameter λ, and normalisation constant α.

Proof:
Follows directly from the previous lemma and the definitions of the compound parameters. ■

The causal message parameter defined

Definition:

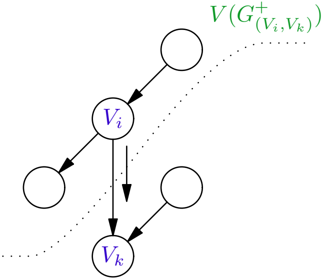
Consider $\mathcal{B} = (G, \Gamma)$ with SCG $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$ and joint $\Pr$.

Let $V_i \in \boldsymbol{V}_G$ have child $V_k \in \boldsymbol{\sigma}(V_i)$

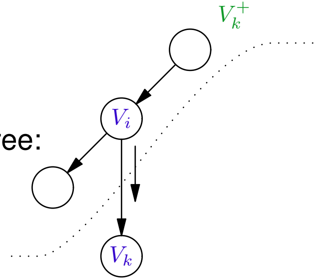
- the function $\pi^{V_i}_{V_k} : \{v_i, \neg v_i\} \to [0, 1]$ is defined by

  $$\pi^{V_i}_{V_k}(V_i) = \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_i,V_k)}}})$$

  and called the causal (message) parameter from V_i to V_k.



directed tree:



The diagnostic message parameter defined

Definition:

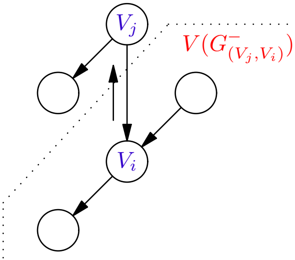
Consider $\mathcal{B} = (G, \Gamma)$ with SCG $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$ and joint $\Pr$.

Let $V_i \in \boldsymbol{V}_G$ have parent $V_j \in \boldsymbol{\rho}(V_i)$;

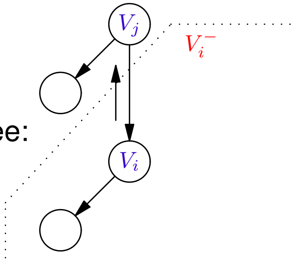
- the function $\lambda_{V_i}^{V_j} : \{v_j, \neg v_j\} \rightarrow [0,1]$ is defined by

$$\lambda_{V_i}^{V_j}(V_j) = \Pr(\tilde{c}_{\boldsymbol{V}_{G^-_{(V_j,V_i)}}} \mid V_j)$$

  and called the diagnostic (message) parameter from V_i to V_j.



directed tree:



Computing compound causal parameters in SCGs

Lemma:

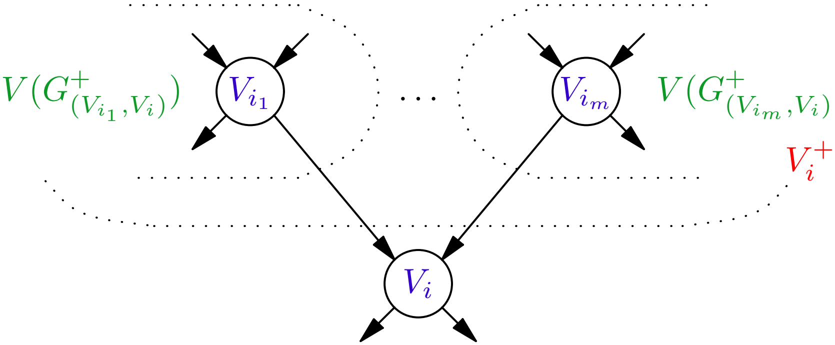
Let $\mathcal{B} = (G, \Gamma)$ be as before. Consider a node $V_i \in \boldsymbol{V}_G$ and its parents $\boldsymbol{\rho}(V_i) = \{V_{i_1}, \ldots, V_{i_m}\}$, $m \geq 1$.

Then

$$\pi(V_i) = \sum_{c_{\boldsymbol{\rho}(V_i)}} \gamma(V_i \mid c_{\boldsymbol{\rho}(V_i)}) \cdot \prod_{j=1,\ldots,m} \pi_{V_i}^{V_{i_j}}(c_{V_{i_j}})$$

where $c_{\boldsymbol{\rho}(V_i)} = \bigwedge_{j=1,\ldots,m} c_{V_{i_j}}$

Note that each $c_{V_{i_j}}$ used in the product should be consistent with the $c_{\boldsymbol{\rho}(V_i)}$ from the summand!



$V(G^{+}_{(V_{i_1},V_i)})$
V_{i_1}
$\cdots$
V_{i_m}
$V(G^{+}_{(V_{i_m},V_i)})$
V_i^{+}
V_i

Computing compound causal parameters in SCGs

Proof:

Let $\Pr$ be the joint distribution defined by $\mathcal{B}$. Then

$$\pi(V_i) \overset{\text{DEF}}{=} \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_i^+}) = \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_{i_1},V_i)}}} \wedge \ldots \wedge \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_{i_m},V_i)}}})$$

$$= \sum_{c_{\boldsymbol{\rho}(V_i)}} \Pr(V_i \mid c_{\boldsymbol{\rho}(V_i)} \wedge \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_{i_1},V_i)}}} \wedge \ldots \wedge \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_{i_m},V_i)}}}) \cdot$$

$$\cdot \Pr(c_{\boldsymbol{\rho}(V_i)} \mid \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_{i_1},V_i)}}} \wedge \ldots \wedge \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_{i_m},V_i)}}})$$

$$= \sum_{c_{\boldsymbol{\rho}(V_i)}} \Pr(V_i \mid c_{\boldsymbol{\rho}(V_i)}) \cdot \prod_{j=1,\ldots,m} \Pr(c_{V_{i_j}} \mid \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_{i_j},V_i)}}})$$

$$= \sum_{c_{\boldsymbol{\rho}(V_i)}} \gamma(V_i \mid c_{\boldsymbol{\rho}(V_i)}) \cdot \prod_{j=1,\ldots,m} \pi^{V_{i_j}}_{V_i}(c_{V_{i_j}})$$

where $c_{\boldsymbol{\rho}(V_i)} = \bigwedge_{j=1,\ldots,m} c_{V_{i_j}}$ ■

Computing π in directed trees

Lemma:

Consider $\mathcal{B} = (G, \Gamma)$ with directed tree G.

Consider a node $V_i \in \boldsymbol{V}_G$ and its parent $\boldsymbol{\rho}(V_i) = \{V_j\}$.

Then

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

Proof:

See the proof for the general case where G is a singly connected graph. Take into account that V_i now only has a single parent V_j. ■

Computing causal message parameters in SCGs

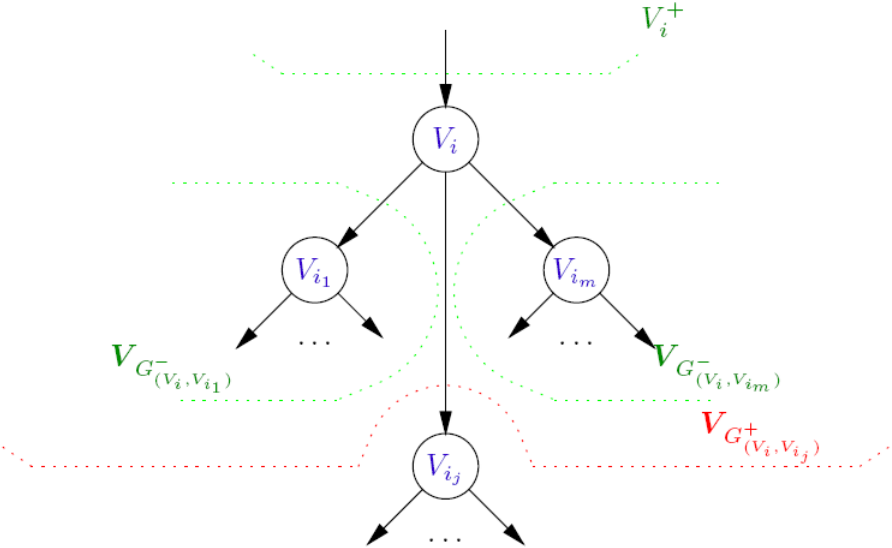
Lemma:

Consider $\mathcal{B} = (G, \Gamma)$ with SCG $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$.
Let $V_i \in \boldsymbol{V}_G$ be an uninstantiated node with $m \geq 1$ children $\boldsymbol{\sigma}(V_i) = \{V_{i_1}, \ldots, V_{i_m}\}$.

Then

$$\pi_{V_{i_j}}^{V_i}(V_i) = \alpha \cdot \pi(V_i) \cdot \prod_{k=1,\ldots,m,\, k \neq j} \lambda_{V_{i_k}}^{V_i}(V_i)$$

where α is a normalisation constant.



V_i^+
V_i
V_{i_1}
V_{i_m}
$V_{G^-_{(V_i,V_{i_1})}}$
$V_{G^-_{(V_i,V_{i_m})}}$
$V_{G^+_{(V_i,V_{i_j})}}$
V_{i_j}

Computing causal message parameters in SCGs

Proof:

Let $\Pr$ be the joint distribution defined by $\mathcal{B}$. Then

$$
\begin{aligned}
\pi^{V_i}_{V_{i_j}}(V_i) &\stackrel{\text{DEF}}{=} \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_{G^+_{(V_i,V_{i_j})}}}) \\
&= \alpha' \cdot \Pr(\widetilde{c}_{\boldsymbol{V}_{G^+_{(V_i,V_{i_j})}}} \mid V_i) \cdot \Pr(V_i) \\
&= \alpha' \cdot \Pr(\widetilde{c}_{\boldsymbol{V}_i^+} \wedge (\bigwedge_{k \neq j} \widetilde{c}_{\boldsymbol{V}_{G^-_{(V_i,V_{i_k})}}}) \mid V_i) \cdot \Pr(V_i) \\
&= \alpha' \cdot \Pr(\widetilde{c}_{\boldsymbol{V}_i^+} \mid V_i) \cdot \prod_{k \neq j} \Pr(\widetilde{c}_{\boldsymbol{V}_{G^-_{(V_i,V_{i_k})}}} \mid V_i) \cdot \Pr(V_i) \\
&= \alpha \cdot \Pr(V_i \mid \widetilde{c}_{\boldsymbol{V}_i^+}) \cdot \prod_{k \neq j} \Pr(\widetilde{c}_{\boldsymbol{V}_{G^-_{(V_i,V_{i_k})}}} \mid V_i) \\
&= \alpha \cdot \pi(V_i) \cdot \prod_{k \neq j} \lambda^{V_i}_{V_{i_k}}(V_i)
\end{aligned}
$$

■

Computing compound diagnostic parameters in SCGs

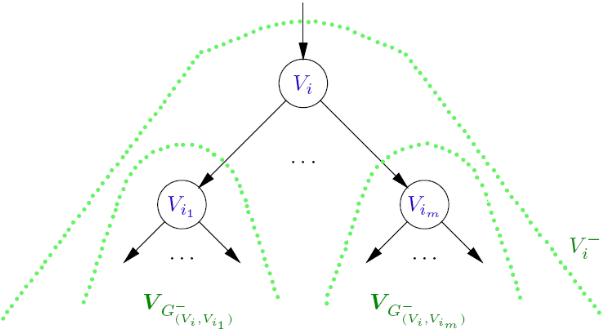
Lemma:

Let $\mathcal{B} = (G, \Gamma)$ be as before.

Consider an uninstantiated node $V_i \in \boldsymbol{V}_G$ with $m \geq 1$ children $\boldsymbol{\sigma}(V_i) = \{V_{i_1}, \ldots, V_{i_m}\}$.

Then

$$\lambda(V_i) = \prod_{j=1,\ldots,m} \lambda_{V_{i_j}}^{V_i}(V_i)$$



Computing compound diagnostic parameters in SCGs

Proof: Let $\Pr$ be the joint distribution defined by $\mathcal{B}$. Then

$$\begin{aligned}
\lambda(V_i) &\stackrel{\text{DEF}}{=} \Pr(\widetilde{c}_{\boldsymbol{V_i^-}} \mid V_i) \\
&= \Pr(\widetilde{c}_{\boldsymbol{V}_{G^-_{(V_i,V_{i_1})}}} \wedge \ldots \wedge \widetilde{c}_{\boldsymbol{V}_{G^-_{(V_i,V_{i_m})}}} \mid V_i) \\
&= \Pr(\widetilde{c}_{\boldsymbol{V}_{G^-_{(V_i,V_{i_1})}}} \mid V_i) \cdot \ldots \cdot \Pr(\widetilde{c}_{\boldsymbol{V}_{G^-_{(V_i,V_{i_m})}}} \mid V_i) \\
&= \lambda^{V_i}_{V_{i_1}}(V_i) \cdot \ldots \cdot \lambda^{V_i}_{V_{i_m}}(V_i) \\
&= \prod_{j=1,\ldots,m} \lambda^{V_i}_{V_{i_j}}(V_i) \qquad \blacksquare
\end{aligned}$$

Computing diagnostic message parameters in SCGs

Lemma:

Let $\mathcal{B} = (G, \Gamma)$ be as before. Consider a node $V_i \in \boldsymbol{V}_G$ with $n \geq 1$ parents $\boldsymbol{\rho}(V_i) = \{V_{j_1}, \ldots, V_{j_n}\}$. Then

$$\lambda_{V_i}^{V_{j_k}}(V_{j_k}) = \alpha \cdot \sum_{c_{V_i}} \lambda(c_{V_i}) \cdot \left[\sum_{x = c_{\boldsymbol{\rho}(V_i) \setminus \{V_{j_k}\}}} \left(\gamma(c_{V_i} \mid x \wedge V_{j_k}) \cdot \prod_{l=1,\ldots,n,\, l \neq k} \pi_{V_i}^{V_{j_l}}(c_{V_{j_l}}) \right) \right]$$

where α is a normalisation constant.

Note that each $c_{V_{j_l}}$ used in the product should be consistent with the x from the summand!

Proof: See syllabus. ■

Computing λ- messages in directed trees

Lemma:

Let $\mathcal{B} = (G, \Gamma)$ be a Bayesian network with directed tree G.

Consider a node $V_i \in \boldsymbol{V}_G$ and its parent $\boldsymbol{\rho}(V_i) = \{V_j\}$.

Then

$$\lambda_{V_i}^{V_j}(V_j) = \sum_{c_{V_i}} \lambda(c_{V_i}) \cdot \gamma(c_{V_i} \mid V_j)$$

Computing λ-messages in directed trees

Proof: Let $\Pr$ be the joint distribution defined by $\mathcal{B}$. Then

$$
\begin{aligned}
\lambda^{V_j}_{V_i}(V_j) &\overset{\mathrm{DEF}}{=} \Pr(\widetilde{c}_{\boldsymbol{V_i}^-} \mid V_j) \\
&= \Pr(\widetilde{c}_{\boldsymbol{V_i}^-} \mid v_i \wedge V_j) \cdot \Pr(v_i \mid V_j) \\
&\quad + \Pr(\widetilde{c}_{\boldsymbol{V_i}^-} \mid \neg v_i \wedge V_j) \cdot \Pr(\neg v_i \mid V_j) \\
&= \Pr(\widetilde{c}_{\boldsymbol{V_i}^-} \mid v_i) \cdot \Pr(v_i \mid V_j) \\
&\quad + \Pr(\widetilde{c}_{\boldsymbol{V_i}^-} \mid \neg v_i) \cdot \Pr(\neg v_i \mid V_j) \\
&= \lambda(v_i) \cdot \gamma(v_i \mid V_j) + \lambda(\neg v_i) \cdot \gamma(\neg v_i \mid V_j) \\
&= \sum_{c_{V_i}} \lambda(c_{V_i}) \cdot \gamma(c_{V_i} \mid V_j) \qquad \blacksquare
\end{aligned}
$$

Pearl's BP inference algorithm: *computation rules*

For $V_i \in \boldsymbol{V}_G$ with $\boldsymbol{\rho}(V_i) = \{V_{j_1}, \ldots, V_{j_n}\}$, $\boldsymbol{\sigma}(V_i) = \{V_{i_1}, \ldots, V_{i_m}\}$:

$$\Pr(V_i \mid \tilde{c}_{\boldsymbol{V}}) = \alpha \cdot \pi(V_i) \cdot \lambda(V_i) \qquad \text{(data fusion)}$$

$$\pi(V_i) = \sum_{c_{\boldsymbol{\rho}(V_i)}} \gamma(V_i \mid c_{\boldsymbol{\rho}(V_i)}) \cdot \prod_{k=1}^{n} \pi_{V_i}^{V_{j_k}}(c_{V_{j_k}})$$

$$\lambda(V_i) = \prod_{j=1}^{m} \lambda_{V_{i_j}}^{V_i}(V_i) \qquad \textit{dummy!}$$

$$\pi_{V_{i_j}}^{V_i}(V_i) = \alpha' \cdot \pi(V_i) \cdot \prod_{k=1, k \neq j}^{m} \lambda_{V_{i_k}}^{V_i}(V_i) \qquad \textit{dummy!}$$

$$\lambda_{V_i}^{V_{j_k}}(V_{j_k}) = \alpha'' \cdot \sum_{c_{V_i}} \lambda(c_{V_i}) \cdot \left[\sum_{x = c_{\boldsymbol{\rho}(V_i) \setminus \{V_{j_k}\}}} \left(\gamma(c_{V_i} \mid x \wedge V_{j_k}) \cdot \prod_{l=1, l \neq k}^{n} \pi_{V_i}^{V_{j_l}}(c_{V_{j_l}}) \right) \right]$$

with normalisation constants α, α', and α''.

Special cases: root nodes

Consider $\mathcal{B} = (G, \Gamma)$ with SCG G and joint distribution $\Pr$.

Consider a node $W \in \boldsymbol{V}_G$ with $\boldsymbol{\rho}(W) = \emptyset$.
The compound causal parameter for W is defined by

$$
\begin{aligned}
\pi(W) &= \Pr(W \mid \widetilde{c}_{\boldsymbol{W}^+}) && \text{(definition)} \\
&= \Pr(W \mid \mathsf{T}) && (\boldsymbol{W}^+ = \emptyset) \\
&= \Pr(W) \\
&= \gamma(W)
\end{aligned}
$$

Special cases: leaf nodes

Let $\mathcal{B} = (G, \Gamma)$ and $\Pr$ be as before.

Consider a node V with $\boldsymbol{\sigma}(V) = \emptyset$.
The compound diagnostic parameter for V is defined as

- if node V is uninstantiated, then

$$\begin{aligned} \lambda(V) &= \Pr(\widetilde{c}_{\boldsymbol{V}^-} \mid V) && \text{(definition)} \\ &= \Pr(\mathsf{T} \mid V) && (\boldsymbol{V}^- = \{V\}, V \text{ uninst.}) \\ &= 1 \end{aligned}$$

- if node V is instantiated, then

$$\begin{aligned} \lambda(V) &= \Pr(\widetilde{c}_{\boldsymbol{V}^-} \mid V) && \text{(definition)} \\ &= \Pr(\widetilde{c}_{\boldsymbol{V}} \mid V) && (\boldsymbol{\sigma}(V) = \emptyset) \\ &= \begin{cases} 1 & \text{for } c_V = \widetilde{c}_{\boldsymbol{V}} \\ 0 & \text{for } c_V \neq \widetilde{c}_{\boldsymbol{V}} \end{cases} \end{aligned}$$

Special cases: uninstantiated (sub)graphs

(Compound) Identity property

Consider a node $V \in \boldsymbol{V}_G$ for which $\tilde{c}_{\boldsymbol{V}^-} =$ T(rue).

- The compound diagnostic parameter for V is defined as:

$$
\begin{aligned}
\lambda(V) &= \Pr(\tilde{c}_{\boldsymbol{V}^-} \mid V) && \text{(definition)} \\
&= \Pr(\mathsf{T} \mid V) && (\tilde{c}_{\boldsymbol{V}^-} = \mathsf{T}) \\
&= 1
\end{aligned}
$$

- If in addition $\tilde{c}_{\boldsymbol{V}_{\boldsymbol{G}^-_{(\boldsymbol{V_p},\boldsymbol{V})}}} =$ T for parent V_p of V, then

$$
\lambda_V^{V_p}(V_p) = \Pr(\tilde{c}_{\boldsymbol{V}_{\boldsymbol{G}^-_{(\boldsymbol{V_p},\boldsymbol{V})}}} \mid V_p) = 1
$$

Both properties trivially hold for all nodes in the prior network.

Special cases: uninstantiated (sub)graphs

Causal parameter equivalence

Consider a node $V \in \boldsymbol{V}_G$, with $\widetilde{c}_{\boldsymbol{V}^-} =$ T(rue), and its child V_k.

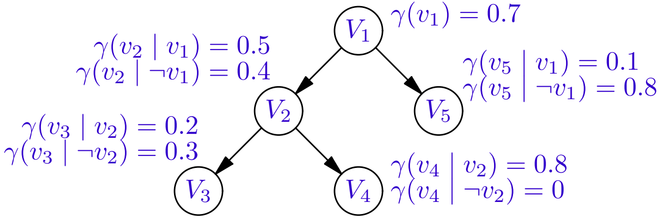
The causal message parameter for V is computed from:

$$\pi^V_{V_k}(V) \;=\; \alpha \cdot \pi(V) \cdot \prod_{i=1, i \neq k}^{m} \lambda^V_{V_i}(V)$$

Since $\widetilde{c}_{\boldsymbol{V}^-} =$ T, we have that for each child V_i of V, $\lambda(V_i) = 1$ and also $\lambda^V_{V_i}(V) = 1$. Hence,

$$\pi^V_{V_k}(V) \;=\; \alpha \cdot \pi(V) \cdot \prod_{i=1, i \neq k}^{m} 1 \;=\; \pi(V)$$

Pearl's BP algorithm: a tree example



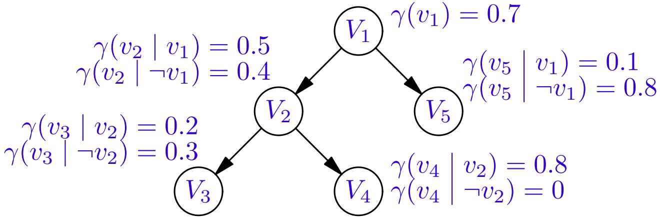
Assignment: compute $\Pr(V_i)$, $i = 1, \ldots, 5$.

Start: $\Pr(V_i) = \alpha \cdot \pi(V_i) \cdot \lambda(V_i)$, $i = 1, \ldots, 5$.

$\lambda(c_{V_i}) = 1$ for all c_{V_i} and V_i. (Identity property)

As a result, no normalisation is required and $\Pr(V_i) = \pi(V_i)$.

An example (2)



$\pi(V_1) = \gamma(V_1)$ (special case: root).

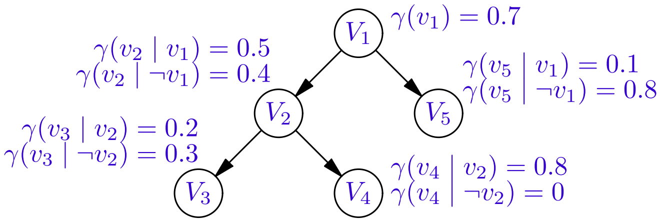
Node V_1 computes:

$$\begin{array}{lllllll} \Pr(v_1) & = & \pi(v_1) & = & \gamma(v_1) & = & 0.7 \\ \Pr(\neg v_1) & = & \pi(\neg v_1) & = & \gamma(\neg v_1) & = & 0.3 \end{array}$$

Node V_1 computes for node V_2:

$$\pi_{V_2}^{V_1}(V_1) = \pi(V_1) \quad \text{(causal parameter equivalence)}$$

An example (3)

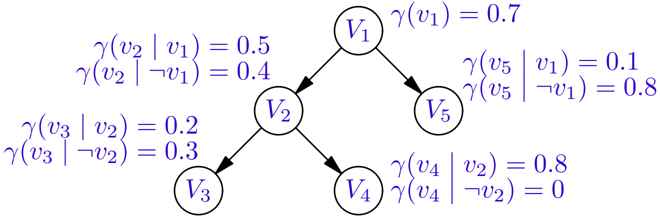


Node V_2 computes:

$$\begin{aligned}
\Pr(v_2) &= \pi(v_2) \\
&= \gamma(v_2 \mid v_1) \cdot \pi_{V_2}^{V_1}(v_1) + \gamma(v_2 \mid \neg v_1) \cdot \pi_{V_2}^{V_1}(\neg v_1) \\
&= \gamma(v_2 \mid v_1) \cdot \pi(v_1) + \gamma(v_2 \mid \neg v_1) \cdot \pi(\neg v_1) \\
&= 0.5 \cdot 0.7 + 0.4 \cdot 0.3 = 0.47
\end{aligned}$$

$$\Pr(\neg v_2) = \pi(\neg v_2) = 0.5 \cdot 0.7 + 0.6 \cdot 0.3 = 0.53$$

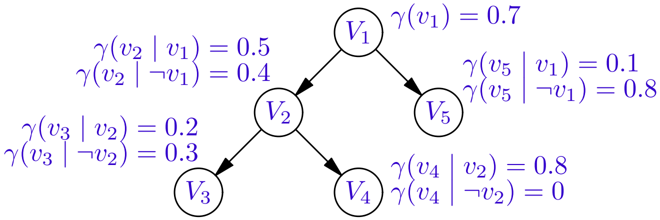
An example (4)



Node V_2 computes for node V_3 and node V_4:

$$\pi_{V_3}^{V_2}(V_2) = \pi_{V_4}^{V_2}(V_2) = \pi(V_2)$$

An example (5)

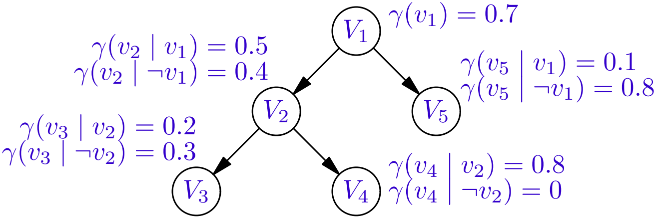


Node V_3 computes:

$$\begin{aligned}
\Pr(v_3) &= \pi(v_3) \\
&= \gamma(v_3 \mid v_2) \cdot \pi_{V_3}^{V_2}(v_2) + \gamma(v_3 \mid \neg v_2) \cdot \pi_{V_3}^{V_2}(\neg v_2) \\
&= \gamma(v_3 \mid v_2) \cdot \pi(v_2) + \gamma(v_3 \mid \neg v_2) \cdot \pi(\neg v_2) \\
&= 0.2 \cdot 0.47 + 0.3 \cdot 0.53 = 0.253
\end{aligned}$$

$$\Pr(\neg v_3) = \pi(\neg v_3) = 0.8 \cdot 0.47 + 0.7 \cdot 0.53 = 0.747$$

An example (6)



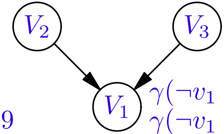
In a similar way, we find that

$$\Pr(v_4) = 0.376, \;\; \Pr(\neg v_4) = 0.624$$
$$\Pr(v_5) = 0.310, \;\; \Pr(\neg v_5) = 0.690$$

■

Pearl's BP algorithm: example in a SCG

$\gamma(v_2) = 0.1$
$\gamma(\neg v_2) = 0.9$



$\gamma(v_3) = 0.4$
$\gamma(\neg v_3) = 0.6$

$$\gamma(v_1 \mid v_2 \wedge v_3) = 0.8$$
$$\gamma(v_1 \mid \neg v_2 \wedge v_3) = 0.9$$
$$\gamma(v_1 \mid v_2 \wedge \neg v_3) = 0.5$$
$$\gamma(v_1 \mid \neg v_2 \wedge \neg v_3) = 0.6$$

$$\gamma(\neg v_1 \mid v_2 \wedge v_3) = 0.2$$
$$\gamma(\neg v_1 \mid \neg v_2 \wedge v_3) = 0.1$$
$$\gamma(\neg v_1 \mid v_2 \wedge \neg v_3) = 0.5$$
$$\gamma(\neg v_1 \mid \neg v_2 \wedge \neg v_3) = 0.4$$

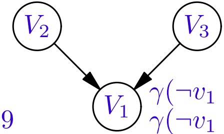
Assignment: compute $\Pr(V_1) = \alpha \cdot \pi(V_1) \cdot \lambda(V_1)$.

$\lambda(v_1) = \lambda(\neg v_1) = 1$ (Identity property)

As a result, no normalisation is required.

An example (2)

$\gamma(v_2) = 0.1$
$\gamma(\neg v_2) = 0.9$



$\gamma(v_3) = 0.4$
$\gamma(\neg v_3) = 0.6$

$\gamma(v_1 \mid v_2 \wedge v_3) = 0.8$
$\gamma(v_1 \mid \neg v_2 \wedge v_3) = 0.9$
$\gamma(v_1 \mid v_2 \wedge \neg v_3) = 0.5$
$\gamma(v_1 \mid \neg v_2 \wedge \neg v_3) = 0.6$

$\gamma(\neg v_1 \mid v_2 \wedge v_3) = 0.2$
$\gamma(\neg v_1 \mid \neg v_2 \wedge v_3) = 0.1$
$\gamma(\neg v_1 \mid v_2 \wedge \neg v_3) = 0.5$
$\gamma(\neg v_1 \mid \neg v_2 \wedge \neg v_3) = 0.4$

Node V_1 computes:

$$\begin{aligned}
\Pr(v_1) = \pi(v_1) &= \gamma(v_1 \mid v_2 \wedge v_3) \cdot \pi_{V_1}^{V_2}(v_2) \cdot \pi_{V_1}^{V_3}(v_3) + \\
&\quad + \gamma(v_1 \mid \neg v_2 \wedge v_3) \cdot \pi_{V_1}^{V_2}(\neg v_2) \cdot \pi_{V_1}^{V_3}(v_3) + \\
&\quad + \gamma(v_1 \mid v_2 \wedge \neg v_3) \cdot \pi_{V_1}^{V_2}(v_2) \cdot \pi_{V_1}^{V_3}(\neg v_3) + \\
&\quad + \gamma(v_1 \mid \neg v_2 \wedge \neg v_3) \cdot \pi_{V_1}^{V_2}(\neg v_2) \cdot \pi_{V_1}^{V_3}(\neg v_3) \\
&= 0.8 \cdot 0.1 \cdot 0.4 + 0.9 \cdot 0.9 \cdot 0.4 + \\
&\quad + 0.5 \cdot 0.1 \cdot 0.6 + 0.6 \cdot 0.9 \cdot 0.6 = 0.71 \\
\Pr(\neg v_1) &= 0.29
\end{aligned}$$

■

Instantiated nodes

Let $\mathcal{B} = (G, \Gamma)$ be a BN with SCG G; let $\Pr$ be as before.

Consider an instantiated node $V \in \boldsymbol{V}_G$, for which evidence $V = true$ is obtained.

- For the compound diagnostic parameter $\lambda : \{v, \neg v\} \rightarrow [0, 1]$ for V we have that

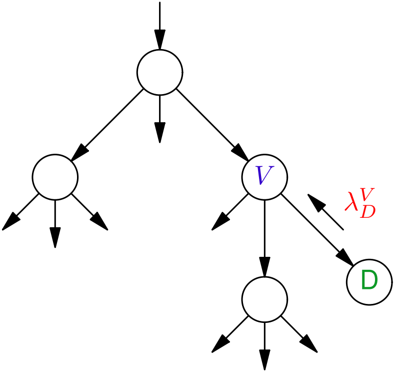
$$\begin{aligned}
\lambda(v) &= \Pr(\widetilde{c}_{\boldsymbol{V}^-} \mid v) && \text{(definition)} \\
&= \Pr(\widetilde{c}_{\boldsymbol{V}^- \setminus \{V\}} \wedge v \mid v) \\
&= ?? \\
&\quad \text{(unless } \boldsymbol{\sigma}(V) = \emptyset \text{ in which case } \lambda(v) = 1)
\end{aligned}$$

$$\begin{aligned}
\lambda(\neg v) &= \Pr(\widetilde{c}_{\boldsymbol{V}^-} \mid \neg v) && \text{(definition)} \\
&= \Pr(\widetilde{c}_{\boldsymbol{V}^- \setminus \{V\}} \wedge v \mid \neg v) \\
&= 0
\end{aligned}$$

The case with evidence $V = false$ is similar.

Entering evidence

Consider a fragment of a BN graph G:



Suppose evidence is obtained for node V.

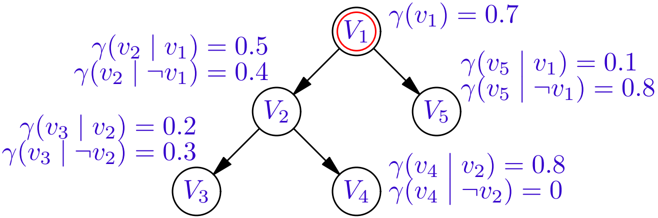
Entering evidence is modelled by extending G with a 'dummy' child D for V.

The dummy node sends the diagnostic parameter λ_D^V to V with

$$\lambda_D^V(v) = 1, \qquad \lambda_D^V(\neg v) = 0 \qquad \text{for evidence } V = \mathit{true}$$

$$\lambda_D^V(v) = 0, \qquad \lambda_D^V(\neg v) = 1 \qquad \text{for evidence } V = \mathit{false}$$

Entering evidence: a tree example



Evidence $V_1 = false$ is entered.

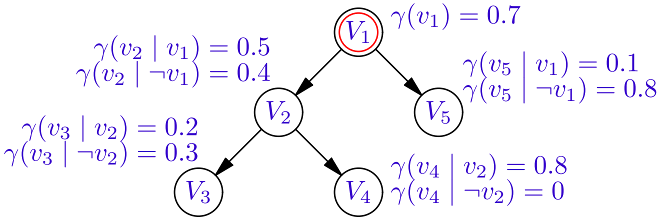
Assignment: compute $\Pr^{\neg v_1}(V_i)$.

Start: $\Pr^{\neg v_1}(V_i) = \alpha \cdot \pi(V_i) \cdot \lambda(V_i)$, $i = 1, \ldots, 5$.

For $i = 2, \ldots, 5$, we have that $\lambda(c_{V_i}) = 1$. (explain why!)

For those nodes we thus have $\Pr(V_i) = \pi(V_i)$.

An example with evidence $V_1 = false$ (2)



Node V_1 now computes:

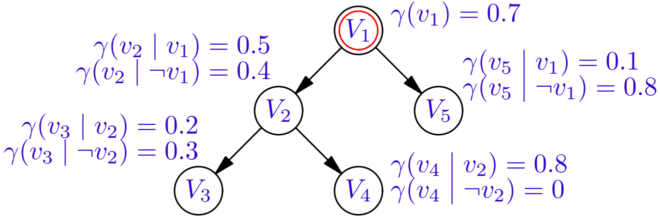
$$\begin{aligned} \Pr^{\neg v_1}(v_1) &= \alpha \cdot \pi(v_1) \cdot \lambda(v_1) &= 0 \\ \Pr^{\neg v_1}(\neg v_1) &= \alpha \cdot \pi(\neg v_1) \cdot \lambda(\neg v_1) &= \alpha \cdot 0.3 \end{aligned}$$

Normalisation gives: $\Pr^{\neg v_1}(v_1) = 0, \quad \Pr^{\neg v_1}(\neg v_1) = 1$

Node V_1 computes for node V_2:

$$\pi_{V_2}^{V_1}(V_1) = \alpha \cdot \pi(V_1) \cdot \lambda_{V_5}^{V_1}(V_1) \cdot \lambda_D^{V_1}(V_1) \quad \Rightarrow 0 \text{ for } \neg v_1, 1 \text{ for } v_1$$

An example with evidence $V_1 = false$ (3)

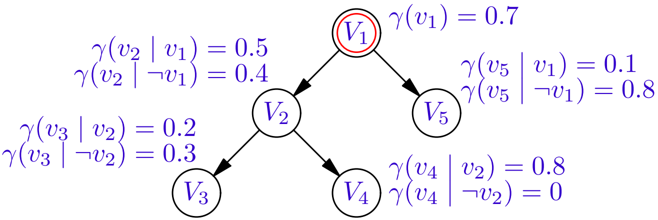


Node V_2 computes:

$$
\begin{aligned}
\Pr^{\neg v_1}(v_2) &= \pi(v_2) \\
&= \gamma(v_2 \mid v_1) \cdot \pi_{V_2}^{V_1}(v_1) + \gamma(v_2 \mid \neg v_1) \cdot \pi_{V_2}^{V_1}(\neg v_1) \\
&= 0.5 \cdot 0 + 0.4 \cdot 1 = 0.4 \\
\Pr^{\neg v_1}(\neg v_2) &= \pi(\neg v_2) = 0.5 \cdot 0 + 0.6 \cdot 1 = 0.6
\end{aligned}
$$

Node V_2 computes for node V_3: $\pi_{V_3}^{V_2}(V_2) = \pi(V_2)$ (explain why!)

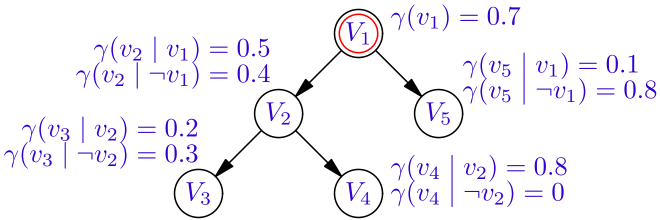
An example with evidence $V_1 = false$ (4)



Node V_3 computes:

$$\begin{aligned}
\Pr^{\neg v_1}(v_3) &= \pi(v_3) \\
&= \gamma(v_3 \mid v_2) \cdot \pi_{V_3}^{V_2}(v_2) + \gamma(v_3 \mid \neg v_2) \cdot \pi_{V_3}^{V_2}(\neg v_2) \\
&= \gamma(v_3 \mid v_2) \cdot \pi(v_2) + \gamma(v_3 \mid \neg v_2) \cdot \pi(\neg v_2) \\
&= 0.2 \cdot 0.4 + 0.3 \cdot 0.6 = 0.26 \\
\Pr^{\neg v_1}(\neg v_3) &= 0.8 \cdot 0.4 + 0.7 \cdot 0.6 = 0.74
\end{aligned}$$

An example with evidence $V_1 = false$ (5)

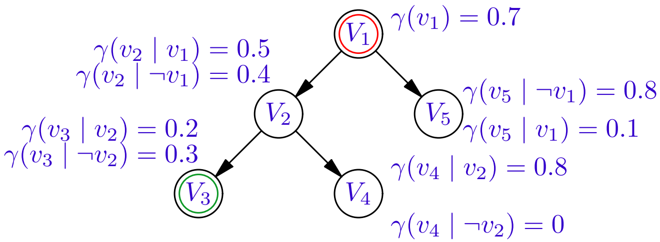


In a similar way, we find that

$$\Pr^{\neg v_1}(v_4) = 0.32, \ \Pr^{\neg v_1}(\neg v_4) = 0.68$$
$$\Pr^{\neg v_1}(v_5) = 0.80, \ \Pr^{\neg v_1}(\neg v_5) = 0.20$$

■

Another piece of evidence: tree example



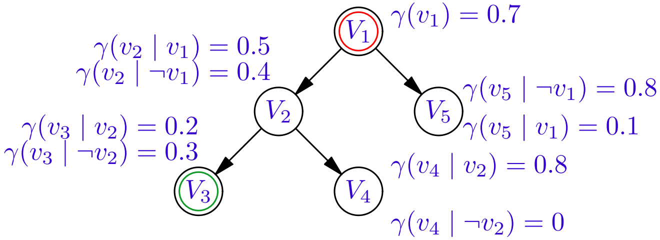
The additional evidence $V_3 = true$ is entered.

Assignment: compute $\Pr^{\neg v_1, v_3}(V_i)$.

Start: $\Pr^{\neg v_1, v_3}(V_i) = \alpha \cdot \pi(V_i) \cdot \lambda(V_i)$, $i = 1, \ldots, 5$.

Which parameters can be re-used? Which need updating?

Another example (2)



For nodes V_i with $i = 4, 5$, $\lambda(c_{V_i}) = 1$ and thus $\Pr(V_i) = \pi(V_i)$.

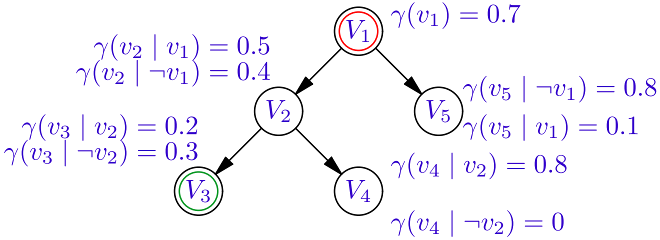
The probabilities for V_1 remain unchanged:

$$\Pr^{\neg v_1, v_3}(v_1) = 0, \quad \Pr^{\neg v_1, v_3}(\neg v_1) = 1$$

The probabilities for node V_5 remain unchanged.
Therefore

$$\Pr^{\neg v_1, v_3}(v_5) = \Pr^{\neg v_1}(\neg v_5) = 0.8, \quad \Pr^{\neg v_1, v_3}(\neg v_5) = 0.2$$

Another example (3)



Node V_3 computes:

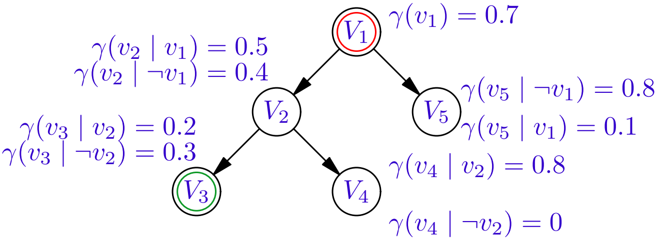
$$\Pr^{\neg v_1, v_3}(v_3) = \alpha \cdot \pi(v_3) \cdot \lambda(v_3) = \alpha \cdot \pi(v_3) = \alpha \cdot 0.26 \cdot 1$$

$$\Pr^{\neg v_1, v_3}(\neg v_3) = \alpha \cdot \pi(\neg v_3) \cdot \lambda(\neg v_3) = 0$$

After normalisation: $\Pr^{\neg v_1, v_3}(v_3) = 1, \; \Pr^{\neg v_1, v_3}(\neg v_3) = 0$

Node V_3 computes for node V_2: $\lambda_{V_3}^{V_2}(V_2) = \sum_{c_{V_3}} \lambda(V_3) \cdot \gamma(c_{V_3} \mid V_2)$

Another example (4)

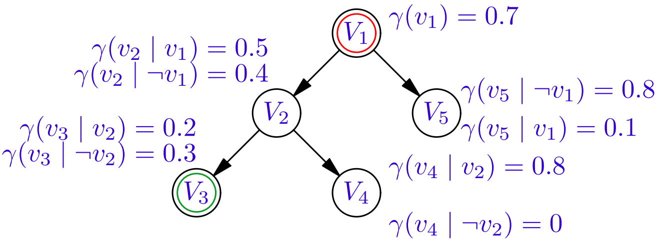


Node V_2 computes:

$$\begin{aligned}
\Pr^{\neg v_1, v_3}(v_2) &= \alpha \cdot \pi(v_2) \cdot \lambda(v_2) = \alpha \cdot \pi(v_2) \cdot \lambda^{V_2}_{V_3}(v_2) \cdot \lambda^{V_2}_{V_4}(v_2) \\
&= \alpha \cdot \pi(v_2) \cdot \gamma(v_3 \mid v_2) = \alpha \cdot 0.4 \cdot 0.2 = \alpha \cdot 0.08 \\
\Pr^{\neg v_1, v_3}(\neg v_2) &= \alpha \cdot \pi(\neg v_2) \cdot \lambda(\neg v_2) = \alpha \cdot \pi(\neg v_2) \cdot \lambda^{V_2}_{V_3}(\neg v_2) \cdot \lambda^{V_2}_{V_4}(\neg v_2) \\
&= \alpha \cdot \pi(\neg v_2) \cdot \gamma(v_3 \mid \neg v_2) = \alpha \cdot 0.6 \cdot 0.3 = \alpha \cdot 0.18
\end{aligned}$$

Normalisation gives: $\Pr^{\neg v_1, v_3}(v_2) = 0.31, \quad \Pr^{\neg v_1, v_3}(\neg v_2) = 0.69$

Another example (5)



Node V_2 computes for node V_4:

$$\pi_{V_4}^{V_2}(V_2) = \alpha \cdot \pi(V_2) \cdot \lambda_{V_3}^{V_2}(V_2) \Rightarrow 0.31 \text{ and } 0.69$$

Node V_4 computes:

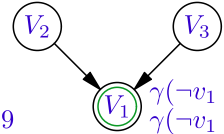
$$\begin{aligned} \Pr{}^{\neg v_1, v_3}(v_4) &= \pi(v_4) = \gamma(v_4 \mid v_2) \cdot \pi_{V_4}^{V_2}(v_2) + \gamma(v_4 \mid \neg v_2) \cdot \pi_{V_4}^{V_2}(\neg v_2) \\ &= \gamma(v_4 \mid v_2) \cdot \pi_{V_4}^{V_2}(v_2) + 0 = 0.8 \cdot 0.31 = 0.248 \end{aligned}$$

$$\Pr{}^{\neg v_1, v_3}(\neg v_4) = 0.2 \cdot 0.31 + 1.0 \cdot 0.69 = 0.752 \quad \blacksquare$$

Entering evidence: example in a SCG

$\gamma(v_2) = 0.1$
$\gamma(\neg v_2) = 0.9$

$\gamma(v_3) = 0.4$
$\gamma(\neg v_3) = 0.6$



$\gamma(v_1 \mid v_2 \wedge v_3) = 0.8$
$\gamma(v_1 \mid \neg v_2 \wedge v_3) = 0.9$
$\gamma(v_1 \mid v_2 \wedge \neg v_3) = 0.5$
$\gamma(v_1 \mid \neg v_2 \wedge \neg v_3) = 0.6$

$\gamma(\neg v_1 \mid v_2 \wedge v_3) = 0.2$
$\gamma(\neg v_1 \mid \neg v_2 \wedge v_3) = 0.1$
$\gamma(\neg v_1 \mid v_2 \wedge \neg v_3) = 0.5$
$\gamma(\neg v_1 \mid \neg v_2 \wedge \neg v_3) = 0.4$

Evidence $V_1 = true$ is entered.

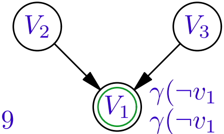
Assignment: compute $\Pr^{v_1}(V_2) = \alpha \cdot \pi(V_2) \cdot \lambda(V_2)$.

$\pi(V_2) = \gamma(V_2)$ (special case : root)

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

An example with evidence $V_1 = true$ (2)

$\gamma(v_2) = 0.1$
$\gamma(\neg v_2) = 0.9$



$\gamma(v_3) = 0.4$
$\gamma(\neg v_3) = 0.6$

$\gamma(v_1 \mid v_2 \wedge v_3) = 0.8$
$\gamma(v_1 \mid \neg v_2 \wedge v_3) = 0.9$
$\gamma(v_1 \mid v_2 \wedge \neg v_3) = 0.5$
$\gamma(v_1 \mid \neg v_2 \wedge \neg v_3) = 0.6$

$\gamma(\neg v_1 \mid v_2 \wedge v_3) = 0.2$
$\gamma(\neg v_1 \mid \neg v_2 \wedge v_3) = 0.1$
$\gamma(\neg v_1 \mid v_2 \wedge \neg v_3) = 0.5$
$\gamma(\neg v_1 \mid \neg v_2 \wedge \neg v_3) = 0.4$

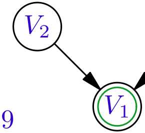
Node V_2 receives from node V_1 [Note: no normalisation!]:

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

$$
\lambda_{V_1}^{V_2}(\neg v_2) = 0.9 \cdot 0.4 + 0.6 \cdot 0.6 = 0.72
$$

An example with evidence $V_1 = true$ (3)

$\gamma(v_2) = 0.1$
$\gamma(\neg v_2) = 0.9$



$\gamma(v_3) = 0.4$
$\gamma(\neg v_3) = 0.6$

$$\begin{aligned}
\gamma(v_1 \mid v_2 \wedge v_3) &= 0.8 \\
\gamma(v_1 \mid \neg v_2 \wedge v_3) &= 0.9 \\
\gamma(v_1 \mid v_2 \wedge \neg v_3) &= 0.5 \\
\gamma(v_1 \mid \neg v_2 \wedge \neg v_3) &= 0.6
\end{aligned}
\qquad
\begin{aligned}
\gamma(\neg v_1 \mid v_2 \wedge v_3) &= 0.2 \\
\gamma(\neg v_1 \mid \neg v_2 \wedge v_3) &= 0.1 \\
\gamma(\neg v_1 \mid v_2 \wedge \neg v_3) &= 0.5 \\
\gamma(\neg v_1 \mid \neg v_2 \wedge \neg v_3) &= 0.4
\end{aligned}$$

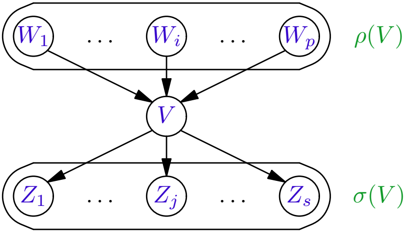
Node V_2 computes:

$$\begin{aligned}
\Pr^{v_1}(v_2) &= \alpha \cdot \pi(v_2) \cdot \lambda(v_2) = \alpha \cdot \gamma(v_2) \cdot \lambda^{V_2}_{V_1}(v_2) = \\
&= \alpha \cdot 0.1 \cdot 0.62 = 0.062\alpha \\
\Pr^{v_1}(\neg v_2) &= \alpha \cdot 0.9 \cdot 0.72 = 0.648\alpha
\end{aligned}$$

Normalisation gives: $\Pr^{v_1}(v_2) \sim 0.087$, $\Pr^{v_1}(\neg v_2) \sim 0.913$ ■

Pearl: some complexity issues

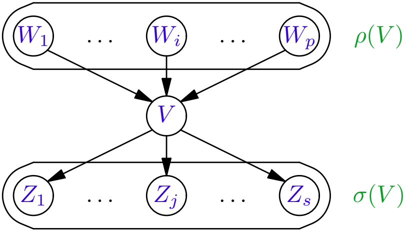
Consider a Bayesian network $\mathcal{B}$ with SCG G with $n \geq 1$ nodes. Suppose node V has p parents and s children:



- Node V computes its compound causal parameter in $O(2^p)$ time:

$$\pi(V) = \sum_{c_{\boldsymbol{\rho}(V)}} \gamma(V \mid c_{\boldsymbol{\rho}(V)}) \cdot \prod_{i=1,\ldots,p} \pi_V^{W_i}(c_{W_i})$$

Complexity issues (2)

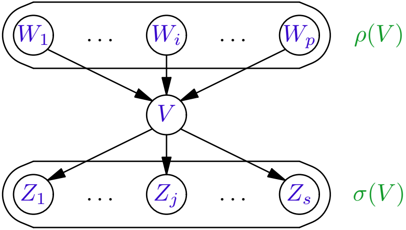


- Computing the compound diagnostic parameter requires $O(s)$ time:

$$\lambda(V) = \prod_{j=1,\ldots,s} \lambda^V_{Z_j}(V)$$

A node can therefore compute the probabilities of its own values in $O(s) + O(2^p)$ time.

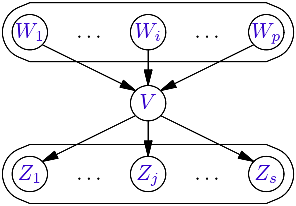
Complexity issues (3)



- Computing a causal message parameter for a child Z_j requires constant time:

$$\pi_{Z_j}^V(V) = \alpha \cdot \pi(V) \cdot \prod_{l=1,\ldots,s,l \neq j} \lambda_{Z_l}^V(V) = \frac{\Pr(V)}{\lambda_{Z_j}^V(V)}$$

Complexity issues (4)



- Computing a diagnostic message parameter for a parent W_i takes $O(2^p)$ time:

$$\lambda_V^{W_i}(W_i) = \alpha \cdot \sum_{c_V} \lambda(c_V) \left[\sum_{c_{\boldsymbol{\rho}(V)\setminus\{W_i\}}} \left(\gamma(V \mid c_{\boldsymbol{\rho}(V)\setminus\{W_i\}} \wedge W_i) \cdot \prod_{l=1,\dots,p,l\neq i} \pi_V^{W_l}(c_{W_l}) \right) \right]$$

A node can compute the messages for all its neighbours in at most $O(s \cdot 1) + O(p \cdot 2^p) = O(p \cdot 2^p)$ time.

If the number of parents per node is bounded by k, then full inference requires at most $O(n \cdot k \cdot 2^k)$ time.

Inference in multiply connected digraphs

When applying Pearl's algorithm to a Bayesian network with a multiply connected digraph, the following problems result:

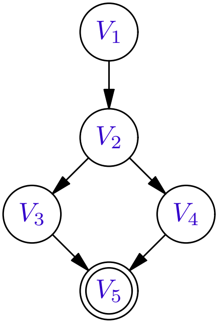
- the message passing does not necessarily reach an equilibrium;
- even if an equilibrium is reached, the computed probabilities are not necessarily correct.

These problems are due to the independences assumed by the BP algorithm, which are invalid in the given Bayesian network.

($\Rightarrow$ approximation algorithm 'Loopy belief propagation')

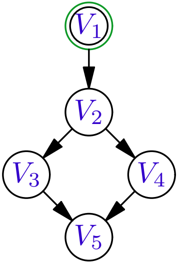
No equilibrium: an example

Consider $\mathcal{B} = (G, \Gamma)$ with multiply connected digraph G:



If node V_5 is instantiated, then the message passing does not necessarily reach an equilibrium.

Incorrect computations: an example (1)



Suppose that evidence $V_1 = true$ is obtained and that we are interested in $\Pr^{v_1}(V_5)$.

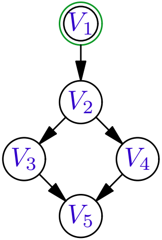
Using marginalisation and independence we find that $\Pr^{v_1}(V_5)$ equals:

$$\Pr^{v_1}(V_5) = \sum_{c_{\{V_2,V_3,V_4\}}} \Pr(V_5 \wedge c_{\{V_2,V_3,V_4\}} \mid v_1)$$

$$= \sum_{c_{\{V_3,V_4\}}} \Pr(V_5 \mid c_{\{V_3,V_4\}}) \cdot \sum_{c_{V_2}} \Pr(c_{V_3} \mid c_{V_2}) \cdot \Pr(c_{V_4} \mid c_{V_2}) \cdot \Pr(c_{V_2} \mid v_1)$$

Note the same value c_{V_2} in the product of the last three terms!

Incorrect computations: an example (2)



Suppose that evidence $V_1 = true$ is obtained and that we are interested in $\Pr^{v_1}(V_5)$.

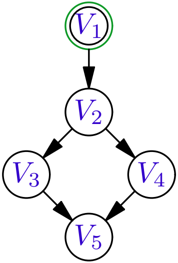
Pearl's algorithm basically computes:

$$\begin{aligned}\Pr^{v_1}(V_5) = & \Pr(V_5 \mid v_3 \wedge v_4) \cdot \Pr(v_3 \mid v_1) \cdot \Pr(v_4 \mid v_1) \\ & + \Pr(V_5 \mid \neg v_3 \wedge v_4) \cdot \Pr(\neg v_3 \mid v_1) \cdot \Pr(v_4 \mid v_1) \\ & + \Pr(V_5 \mid v_3 \wedge \neg v_4) \cdot \Pr(v_3 \mid v_1) \cdot \Pr(\neg v_4 \mid v_1) \\ & + \Pr(V_5 \mid \neg v_3 \wedge \neg v_4) \cdot \Pr(\neg v_3 \mid v_1) \cdot \Pr(\neg v_4 \mid v_1)\end{aligned}$$

and

$$\Pr(V_3 \mid v_1) = \Pr(V_3 \mid v_2) \cdot \Pr(v_2 \mid v_1) + \Pr(V_3 \mid \neg v_2) \cdot \Pr(\neg v_2 \mid v_1)$$
$$\Pr(V_4 \mid v_1) = \Pr(V_4 \mid v_2) \cdot \Pr(v_2 \mid v_1) + \Pr(V_4 \mid \neg v_2) \cdot \Pr(\neg v_2 \mid v_1)$$

Incorrect computations: an example (3)

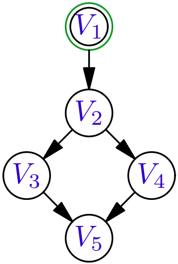


Suppose that evidence $V_1 = true$ is obtained and that we are interested in $\Pr^{v_1}(V_5)$.

Substitution of $\Pr(V_3 \mid v_1)$ and $\Pr(V_4 \mid v_1)$ thus results in incorrect terms, such as for example

$$\Pr(v_5 \mid v_3 \wedge v_4) \cdot \Pr(v_3 \mid v_2) \cdot \Pr(v_2 \mid v_1) \cdot \Pr(v_4 \mid \neg v_2) \cdot \Pr(\neg v_2 \mid v_1)$$

Correct computations: an example



Suppose that evidence $V_1 = true$ is obtained and that we are interested in $\Pr^{v_1}(V_5)$.

This can be computed by conditioning on V_2:

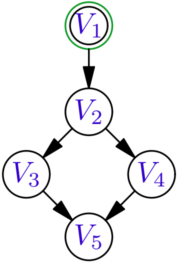
$$\Pr^{v_1}(V_5) = \Pr(V_5 \mid v_2 \wedge v_1) \cdot \Pr(v_2 \mid v_1) + \\ + \Pr(V_5 \mid \neg v_2 \wedge v_1) \cdot \Pr(\neg v_2 \mid v_1)$$

Pearl's algorithm can correctly compute: $\Pr^{v_1}(V_5 \mid V_2)$, e.g.:

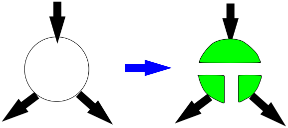
$$\begin{aligned}\Pr^{v_1}(V_5 \mid v_2) = & \Pr(V_5 \mid v_3 \wedge v_4) \cdot \Pr(v_3 \mid v_2 \wedge v_1) \cdot \Pr(v_4 \mid v_2 \wedge v_1) + \\ & \Pr(V_5 \mid \neg v_3 \wedge v_4) \cdot \Pr(\neg v_3 \mid v_2 \wedge v_1) \cdot \Pr(v_4 \mid v_2 \wedge v_1) + \\ & \Pr(V_5 \mid v_3 \wedge \neg v_4) \cdot \Pr(v_3 \mid v_2 \wedge v_1) \cdot \Pr(\neg v_4 \mid v_2 \wedge v_1) + \\ & \Pr(V_5 \mid \neg v_3 \wedge \neg v_4) \cdot \Pr(\neg v_3 \mid v_2 \wedge v_1) \cdot \Pr(\neg v_4 \mid v_2 \wedge v_1)\end{aligned}$$

Compare: $\Pr^{v_1,v_2}(V_5) = \sum_{c_{\{V_3,V_4\}}} \Pr(V_5 \wedge c_{\{V_3,V_4\}} \mid v_1 \wedge v_2)$

An example



When node V_2 is instantiated, digraph G behaves as a SCG:



A solution: Cutset Conditioning

The idea behind cutset conditioning for computing $\Pr(V \mid \tilde{c}_{\boldsymbol{V_G}})$:

1. Select a loop cutset of G:

   nodes $\boldsymbol{L}_G \subseteq \boldsymbol{V}_G$ such that instantiating $\boldsymbol{L}_G$ makes the digraph 'behave' as if it were singly connected;

2. Compute $\Pr(V \mid \tilde{c}_{\boldsymbol{V_G}} \wedge c_{\boldsymbol{L}_G})$ for all possible loop cutset configurations $c_{\boldsymbol{L}_G}$;

3. Marginalise out (= sum out) the loop cutset node(s) $\boldsymbol{L}_G$.

A loop cutset

Definition: Let $G = (\boldsymbol{V}_G, \boldsymbol{A}_G)$ be an acyclic digraph.

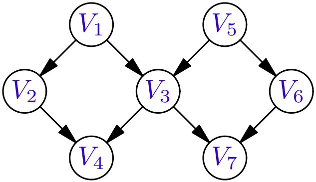
A set $\boldsymbol{L}_G \subseteq \boldsymbol{V}_G$ is called a loop cutset of G if:

every simple cyclic chain (loop) s in G contains a node X such that:

- $X \in \boldsymbol{L}_G$, and
- X has at most one incoming arc on s.

NB a cyclic chain (loop) is not a cycle; a cycle is defined as a cyclic *path*!

An example: loop cutsets



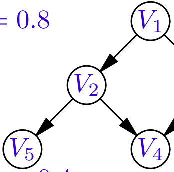
- How many loops does G contain ?
- Which of the following sets are loop cutsets of G ?:
  - $\emptyset$
  - $\{V_1\}$
  - $\{V_3\}$ ✓
  - $\{V_1, V_5\}$ ✓
  - $\{V_2, V_7\}$
  - $\{V_4, V_7\}$
  - $\{V_1, V_2, V_3\}$ ✓
  - $\{V_1, V_4, V_5, V_6, V_7\}$ ✓

Pearl with cutset conditioning: an example (1)

Consider $\mathcal{B} = (G, \Gamma)$ with multiply connected digraph G:

$\gamma(v_1) = 0.8$

$\gamma(v_2 \mid v_1) = 0.9$
$\gamma(v_2 \mid \neg v_1) = 0.3$



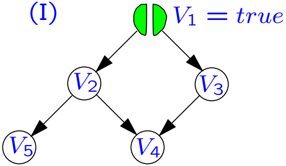
$\gamma(v_3 \mid v_1) = 0.2$
$\gamma(v_3 \mid \neg v_1) = 0.6$

$\gamma(v_5 \mid v_2) = 0.4$
$\gamma(v_5 \mid \neg v_2) = 0.5$

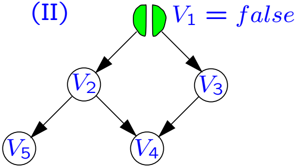
$\gamma(v_4 \mid v_2 \wedge v_3) = 0.1$
$\gamma(v_4 \mid \neg v_2 \wedge v_3) = 0.2$
$\gamma(v_4 \mid v_2 \wedge \neg v_3) = 0.6$
$\gamma(v_4 \mid \neg v_2 \wedge \neg v_3) = 0.1$

We are interested in the probabilities $\Pr(v_4)$ and $\Pr(\neg v_4)$. We choose $\boldsymbol{L}_G = \{V_1\}$. Pearl's algorithm is now applied twice:

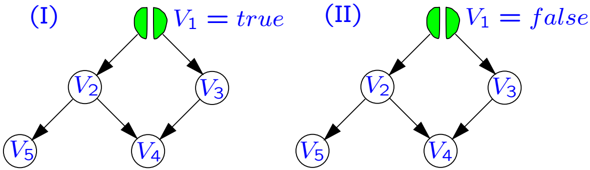
(I)



(II)



Pearl with cutset conditioning: example (2: general)



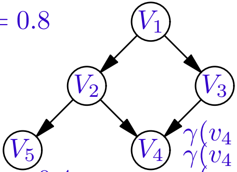
Pearl applied to (I) gives $\Pr(v_4 \mid v_1)$ and $\Pr(\neg v_4 \mid v_1)$;
Pearl applied to (II) gives $\Pr(v_4 \mid \neg v_1)$ and $\Pr(\neg v_4 \mid \neg v_1)$.

The probabilities of interest are finally computed using marginalisation (probability theory):

$$\begin{aligned}\Pr(v_4) &= \Pr(v_4 \mid v_1)\cdot\Pr(v_1) + \Pr(v_4 \mid \neg v_1)\cdot\Pr(\neg v_1)\\ \Pr(\neg v_4) &= \Pr(\neg v_4 \mid v_1)\cdot\Pr(v_1) + \Pr(\neg v_4 \mid \neg v_1)\cdot\Pr(\neg v_1)\end{aligned}$$

where $\Pr(v_1) = 0.8,\ \Pr(\neg v_1) = 0.2$ are the *prior* probabilities for node V_1 (not conditioned on loop cutset configurations!)

Pearl with cutset conditioning: example (3: in detail)



$$\gamma(v_1) = 0.8$$

$$\gamma(v_2 \mid v_1) = 0.9 \qquad \gamma(v_3 \mid v_1) = 0.2$$
$$\gamma(v_2 \mid \neg v_1) = 0.3 \qquad \gamma(v_3 \mid \neg v_1) = 0.6$$

$$\gamma(v_4 \mid v_2 \wedge v_3) = 0.1$$
$$\gamma(v_4 \mid \neg v_2 \wedge v_3) = 0.2$$
$$\gamma(v_4 \mid v_2 \wedge \neg v_3) = 0.6$$
$$\gamma(v_4 \mid \neg v_2 \wedge \neg v_3) = 0.1$$

$$\gamma(v_5 \mid v_2) = 0.4$$
$$\gamma(v_5 \mid \neg v_2) = 0.5$$

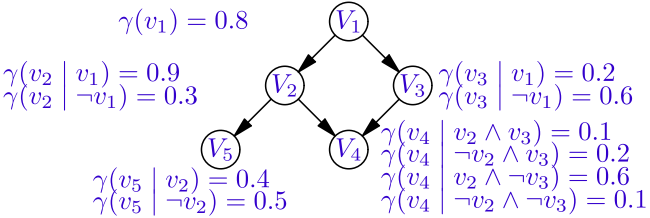
Pearl applied to situation (I) where $V_1 = true$:

$$\Pr(v_4 \mid v_1) = \Pr^{v_1}(v_4) \qquad = \alpha \cdot \pi(v_4) \cdot \lambda(v_4) = \pi(v_4)$$
$$\Pr(\neg v_4 \mid v_1) = \Pr^{v_1}(\neg v_4) = \pi(\neg v_4)$$

The compound causal parameter is computed:

$$\begin{aligned}\pi(v_4) = {} & \gamma(v_4 \mid v_2 \wedge v_3) \cdot \pi_{V_4}^{V_2}(v_2) \cdot \pi_{V_4}^{V_3}(v_3) + \\ & \gamma(v_4 \mid \neg v_2 \wedge v_3) \cdot \pi_{V_4}^{V_2}(\neg v_2) \cdot \pi_{V_4}^{V_3}(v_3) + \\ & \gamma(v_4 \mid v_2 \wedge \neg v_3) \cdot \pi_{V_4}^{V_2}(v_2) \cdot \pi_{V_4}^{V_3}(\neg v_3) + \\ & \gamma(v_4 \mid \neg v_2 \wedge \neg v_3) \cdot \pi_{V_4}^{V_2}(\neg v_2) \cdot \pi_{V_4}^{V_3}(\neg v_3) = \ldots\end{aligned}$$

Pearl with cutset conditioning: example (4)

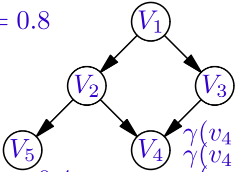


...

$$\pi(v_4) = 0.1 \cdot 0.9 \cdot 0.2 + 0.2 \cdot 0.1 \cdot 0.2 + \\ + 0.6 \cdot 0.9 \cdot 0.8 + 0.1 \cdot 0.1 \cdot 0.8 = 0.462$$

Similarly, we find $\pi(\neg v_4) = 0.538$

Pearl with cutset conditioning: example (5)



$\gamma(v_1) = 0.8$

$\gamma(v_2 \mid v_1) = 0.9$
$\gamma(v_2 \mid \neg v_1) = 0.3$

$\gamma(v_3 \mid v_1) = 0.2$
$\gamma(v_3 \mid \neg v_1) = 0.6$

$\gamma(v_4 \mid v_2 \wedge v_3) = 0.1$
$\gamma(v_4 \mid \neg v_2 \wedge v_3) = 0.2$
$\gamma(v_4 \mid v_2 \wedge \neg v_3) = 0.6$
$\gamma(v_4 \mid \neg v_2 \wedge \neg v_3) = 0.1$

$\gamma(v_5 \mid v_2) = 0.4$
$\gamma(v_5 \mid \neg v_2) = 0.5$

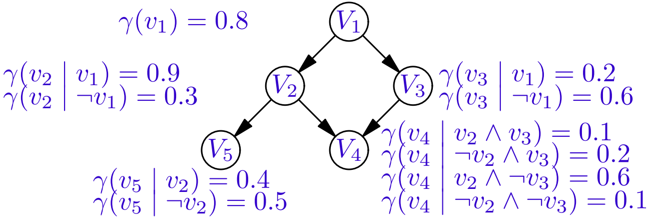
Pearl applied to situation (II) where $V_1 = false$:

$$\Pr(v_4 \mid \neg v_1) = \alpha \cdot \pi(v_4) \cdot \lambda(v_4) = \pi(v_4)$$
$$\Pr(\neg v_4 \mid \neg v_1) = \pi(\neg v_4)$$

where

$$\begin{aligned}\pi(v_4) = {} & \gamma(v_4 \mid v_2 \wedge v_3) \cdot \pi^{V_2}_{V_4}(v_2) \cdot \pi^{V_3}_{V_4}(v_3) + \\ & \gamma(v_4 \mid \neg v_2 \wedge v_3) \cdot \pi^{V_2}_{V_4}(\neg v_2) \cdot \pi^{V_3}_{V_4}(v_3) + \\ & \gamma(v_4 \mid v_2 \wedge \neg v_3) \cdot \pi^{V_2}_{V_4}(v_2) \cdot \pi^{V_3}_{V_4}(\neg v_3) + \\ & \gamma(v_4 \mid \neg v_2 \wedge \neg v_3) \cdot \pi^{V_2}_{V_4}(\neg v_2) \cdot \pi^{V_3}_{V_4}(\neg v_3) \ = \ldots\end{aligned}$$

Pearl with cutset conditioning: example (6)



...

$$\pi(v_4) = 0.1 \cdot 0.3 \cdot 0.6 + 0.2 \cdot 0.7 \cdot 0.6 + \\ + 0.6 \cdot 0.3 \cdot 0.4 + 0.1 \cdot 0.7 \cdot 0.4 = 0.202$$

Similarly, we find $\pi(\neg v_4) = 0.798$

Pearl with cutset conditioning: example completed

Recall: we are interested in $\Pr(v_4)$ and $\Pr(\neg v_4)$.
With Pearl's algorithm we computed

$$\begin{aligned}
\Pr(v_4 \mid v_1) &= 0.462\\
\Pr(\neg v_4 \mid v_1) &= 0.538\\
\Pr(v_4 \mid \neg v_1) &= 0.202\\
\Pr(\neg v_4 \mid \neg v_1) &= 0.798
\end{aligned}$$

From the assessment functions we establish that

$$\Pr(v_1) = 0.8,\ \ \Pr(\neg v_1) = 0.2$$

Resulting in (marginalisation)

$$\begin{aligned}
\Pr(v_4) &= \Pr(v_4 \mid v_1) \cdot \Pr(v_1) + \Pr(v_4 \mid \neg v_1) \cdot \Pr(\neg v_1)\\
&= 0.462 \cdot 0.8 + 0.202 \cdot 0.2 = 0.41\\
\Pr(\neg v_4) &= \Pr(\neg v_4 \mid v_1) \cdot \Pr(v_1) + \Pr(\neg v_4 \mid \neg v_1) \cdot \Pr(\neg v_1)\\
&= 0.538 \cdot 0.8 + 0.798 \cdot 0.2 = 0.59 \qquad \blacksquare
\end{aligned}$$

Cutset conditioning with evidence $\tilde{c}_{V_G}$

Let $\boldsymbol{L}_G$ be a loop cutset for digraph G. Then cutset conditioning exploits that for all $V_i \in \boldsymbol{V}_G$:

$$\Pr(V_i \mid \tilde{c}_{\boldsymbol{V}_G}) = \sum_{c_{\boldsymbol{L}_G}} \underbrace{\Pr(V_i \mid \tilde{c}_{\boldsymbol{V}_G} \wedge c_{\boldsymbol{L}_G})}_{\text{Pearl (from } \mathcal{B})} \cdot \underbrace{\Pr(c_{\boldsymbol{L}_G} \mid \tilde{c}_{\boldsymbol{V}_G})}_{\text{recursively}}$$

<u>Recursion: step 1</u> for 1-st piece of evidence e_1:

$$\Pr(c_{\boldsymbol{L}_G} \mid e_1) = \alpha \cdot \underbrace{\Pr(e_1 \mid c_{\boldsymbol{L}_G})}_{\text{Pearl (from } \mathcal{B})} \cdot \underbrace{\Pr(c_{\boldsymbol{L}_G})}_{\text{marginalisation (from } \Pr!)}$$

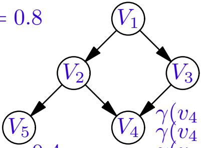
<u>Recursion: step j</u>

$$\Pr(c_{\boldsymbol{L}_G} \mid e_1 \wedge \ldots \wedge e_j) = \alpha \cdot \underbrace{\Pr(e_j \mid c_{\boldsymbol{L}_G} \wedge e_1 \wedge \ldots \wedge e_{j-1})}_{\text{Pearl (from } \mathcal{B})} \cdot \underbrace{\Pr(c_{\boldsymbol{L}_G} \mid e_1 \wedge \ldots \wedge e_{j-1})}_{\text{Step } j-1}$$

An example: cutset conditioning with evidence

$\gamma(v_1) = 0.8$

$\gamma(v_2 \mid v_1) = 0.9$
$\gamma(v_2 \mid \neg v_1) = 0.3$



$\gamma(v_3 \mid v_1) = 0.2$
$\gamma(v_3 \mid \neg v_1) = 0.6$

$\gamma(v_4 \mid v_2 \wedge v_3) = 0.1$
$\gamma(v_4 \mid \neg v_2 \wedge v_3) = 0.2$
$\gamma(v_4 \mid v_2 \wedge \neg v_3) = 0.6$
$\gamma(v_4 \mid \neg v_2 \wedge \neg v_3) = 0.1$

$\gamma(v_5 \mid v_2) = 0.4$
$\gamma(v_5 \mid \neg v_2) = 0.5$

Use loop cutset $\{V_1\}$. Initially we have loop cutset configurations: $\Pr(v_1) = 0.8$ and $\Pr(\neg v_1) = 0.2$.

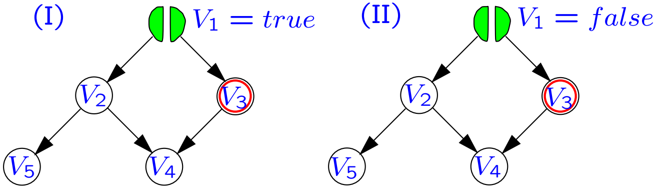
Let's process evidence $V_3 = false$. Updated probabilities are now established for the loop cutset configurations:

$$\Pr^{\neg v_3}(v_1) = \alpha \cdot \underbrace{\Pr(\neg v_3 \mid v_1)}_{\text{Pearl}} \cdot \underbrace{\Pr(v_1)}_{\text{old}} = \alpha \cdot 0.8 \cdot 0.8 = \alpha \cdot 0.64 \Rightarrow 0.89$$

$$\Pr^{\neg v_3}(\neg v_1) = \alpha \cdot \Pr(\neg v_3 \mid \neg v_1) \cdot \Pr(\neg v_1) = \alpha \cdot 0.4 \cdot 0.2 = \alpha \cdot 0.08 \Rightarrow 0.11$$

An example (2)

We are interested in $\Pr^{\neg v_3}(v_4)$. Pearl's algorithm is applied twice:



$\Pr(v_4 \mid v_1 \wedge \neg v_3) = 0.55$
$\Pr(\neg v_4 \mid v_1 \wedge \neg v_3) = 0.45$

$\Pr(v_4 \mid \neg v_1 \wedge \neg v_3) = 0.25$
$\Pr(\neg v_4 \mid \neg v_1 \wedge \neg v_3) = 0.75$

Recall that $\Pr^{\neg v_3}(v_1) = 0.89$, $\Pr^{\neg v_3}(\neg v_1) = 0.11$. Now:

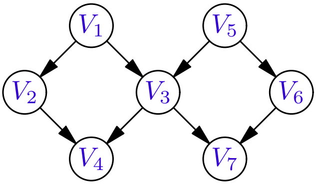
$$\begin{aligned}\Pr^{\neg v_3}(v_4) &= \Pr(v_4 \mid v_1 \wedge \neg v_3) \cdot \Pr(v_1 \mid \neg v_3) \\ &\quad + \Pr(v_4 \mid \neg v_1 \wedge \neg v_3) \cdot \Pr(\neg v_1 \mid \neg v_3) \\ &= 0.55 \cdot 0.89 + 0.25 \cdot 0.11 = 0.52 \quad \blacksquare\end{aligned}$$

Minimal and optimal loop cutsets

Definition: A loop cutset L_G for acyclic digraph G is called

- minimal: if no proper subset $L \subset L_G$ is a loop cutset for G;
- optimal: if for all loop cutsets $L'_G \neq L_G$ for G: $|L'_G| \geq |L_G|$.

Example: Consider the following acyclic digraph G:



Which of the following loop cutsets for G are *minimal*; which are *optimal*? $\{V_3\}$✓✓, $\{V_1, V_3\}$✓, $\{V_1, V_5\}$

Finding an optimal loop cutset

Lemma: The problem of finding an optimal loop cutset for an acyclic digraph is NP-hard.

Proof: The property can be proven by reduction from the "Minimal Vertex Cover"-Problem. For details, see

> *H.J. Suermondt, G.F. Cooper (1990). Probabilistic inference in multiply connected belief networks using loop cutsets, International Journal of Approximate Reasoning, vol. 4, pp. 283 – 306.*

■

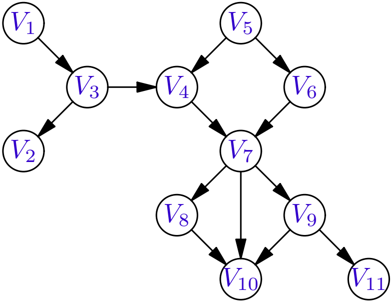
A heuristic algorithm (Suermondt & Cooper)

The following algorithm is a heuristic for finding an optimal loop cutset for a given acyclic digraph G:

PROCEDURE LOOP-CUTSET(G, $\boldsymbol{L}_G$):

WHILE THERE ARE NODES IN G DO
    IF THERE IS A NODE $V_i \in \boldsymbol{V}_G$ WITH $degree(V_i) \leq 1$
    THEN SELECT NODE V_i
    ELSE DETERMINE ALL NODES $\boldsymbol{K} = \{V \in \boldsymbol{V}_G \mid indegree(V) \leq 1\}$
            (THE CANDIDATES FOR THE LOOP CUTSET);
        SELECT A CANDIDATE NODE $V_i \in \boldsymbol{K}$ WITH
            $degree(V_i) \geq degree(V)$ FOR ALL OTHER $V \in \boldsymbol{K}$;
        ADD NODE V_i TO THE LOOP CUTSET $\boldsymbol{L}_G$
    FI;
    DELETE NODE V_i AND ITS INCIDENT ARCS FROM G
OD;
END

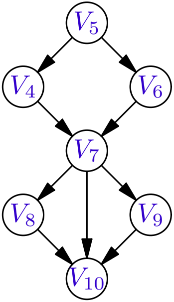
An example



(Recursively) deleting all nodes V_i with $degree(V_i) \leq 1$ gives . . .

An example (2)

(Recursively) deleting all nodes V_i with $degree(V_i) \leq 1$ gives:

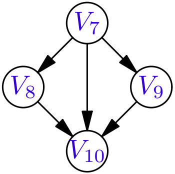


The following nodes are candidates for the loop cutset: V_4, V_5, V_6, V_8, V_9. All have degree 2.

Suppose that node V_4 is selected and added to the loop cutset...

An example (3)

After deleting node V_4 and recursively deleting all remaining V_i with $degree(V_i) \leq 1$ we get:



The following nodes are candidates for the loop cutset: V_7, V_8, V_9.

Node V_7 has highest degree (3) and is selected for the loop cutset.

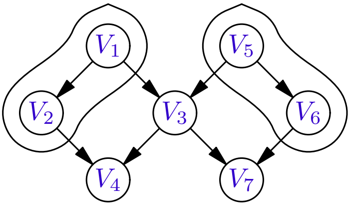
After deleting node V_7 and recursively deleting all remaining nodes V_i with $degree(V_i) \leq 1$ the empty graph results.

The loop cutset found is $\{V_4, V_7\}$. There are other possibilities!

Some properties of the heuristic algorithm

- it always finds a loop cutset for a given acyclic digraph;
- it does not always find an optimal loop cutset;

Example: Consider the following graph G:



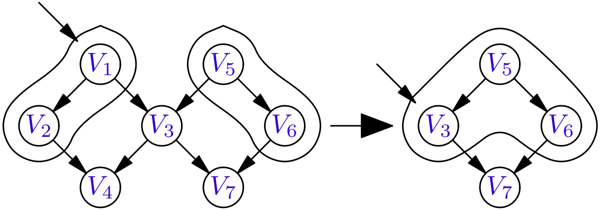
What is the optimal loop cutset for G ? Why won't the algorithm find this loop cutset ? ■

- it found an optimal loop cutset for 70% of the graphs randomly generated in an experiment.

Some properties – continued

- the heuristic does not always find a minimal loop cutset.

Example: Reconsider graph G:



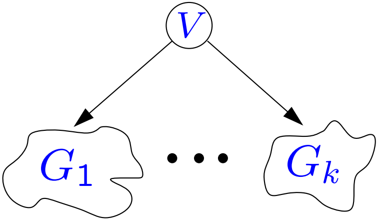
The algorithm could, for example, return the loop cutset $\{V_1, V_3\}$ for G; this loop cutset is not minimal. ■

Note that this problem can be easily resolved afterwards.

Some properties – continued

- the heuristic can select nodes for the loop cutset that are not on a cyclic chain.

  Example:
  Consider the following graph G, where $G_1, \ldots, G_k$, $k >> 1$, are non-singly connected graphs:

  The algorithm can select node V for addition to the loop cutset. ■

  This can be similarly resolved.

Pearl: complexity issues

Consider a Bayesian network $\mathcal{B} = (G, \Gamma)$.

- Let G be a singly connected digraph with n nodes $V_i \in \boldsymbol{V}_G$.

  If $|\boldsymbol{\rho}(V_i)|$ in G is bounded by a small constant, then computing the probabilities for V_i costs time linear in n.

- Let G be a multiply connected digraph with n nodes $V_i \in \boldsymbol{V}_G$ and let $\boldsymbol{L}_G$ be a loop cutset for G.

  If Pearl's algorithm is used in combination with loop cutset conditioning, then all calculations are repeated $2^{|\boldsymbol{L}_G|}$ times.

Summary Pearl: idea and complexity

Idea of Pearl extended with loop cutset conditioning:

1. condition on loop cutset $\rightarrow$ multiply connected graph behaves singly connected
2. update probabilities by message-passing between nodes (= 'standard' Pearl)
3. marginalise out loop cutset

Complexity for all $\Pr(V_i \mid c_{\boldsymbol{E}})$ simultaneously:

- singly connected graphs: $O(n \cdot k \cdot \exp(k))$, where $k = \max_{V_i} |\boldsymbol{\rho}_G(V_i)|$
- multiply connected graphs: $O(n \cdot k \cdot \exp(k + l))$, where $l = |\boldsymbol{L}_G|$

Probabilistic inference: complexity issues

- In general, probabilistic inference with an arbitrary Bayesian network is NP-hard;

  G.F. Cooper (1990). The computational complexity of probabilistic inference using Bayesian belief networks, Artificial Intelligence, vol. 42, pp. 393 – 405.

  This even holds for approximation algorithms, such as e.g. *loopy propagation*!
- all existing algorithms for probabilistic inference have an exponential worst-case complexity;
- the existing algorithms for probabilistic inference have a polynomial time complexity for certain types of Bayesian network ($\sim$ the sparser the graph, the better).

Probabilistic models including continuous variables

Our definition of Bayesian network assumes all variables in γ_V to be discrete.

- this typical assumption can be relaxed[5];
- $\sum$ for discrete variable $\rightarrow$ $\int$ for continuous variable;
- exact inference is possible for a restricted family of distributions (conjugate exponential, e.g. Gaussian);

  methods are similar to those for discrete case;
  (See slide 108)
- otherwise only approximate inference is possible.
  (See slide 109)

---

[5]More on hybrid BNs? See Coursera lecture, and Salmerón et al. 'A Review of Inference Algorithms for Hybrid Bayesian Networks' in JAIR 2018