Probabilistic Programming

Lecture #18: Bayesian Networks

Joost-Pieter Katoen

Software Modeling and Verification Chair | RWTH AACHEN UNIVERSITY

RWTH Lecture Series on Probabilistic Programming 2018

Overview

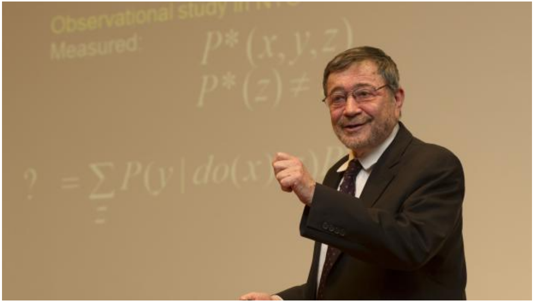
Overview

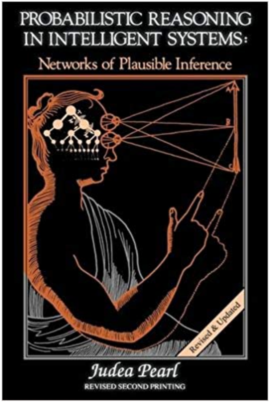
The importance of Bayesian networks

"Bayesian networks are as important to AI and machine learning as Boolean circuits are to computer science."

[Stuart Russell (Univ. of California, Berkeley), 2009]

Judea Pearl: The father of Bayesian networks





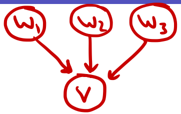
Turing Award 2011: "for fundamental contributions to AI through the development of a calculus for probabilistic and causal reasoning".

Probabilistic graphical models

- Combine graph theory and probability theory
  - Vertices are random variables
  - Edges are dependencies between these variables
  - Enable usage of graph algorithms
  - Graph representation makes (conditional) independence explicit
- Two main types of probabilistic graphical models
  - directed acyclic graphs: Bayesian networks
  - undirected graphs: Markov random fields
- We consider only discrete random variables

Overview

Bayesian networks



Bayesian network

A Bayesian network (BN, for short) is a tuple $B = (V, E, \Theta)$ where

- (V, E) is a directed acyclic graph with finite V in which each $v \in V$ represents a random variable with values from finite domain D, and $(v, w) \in E$ represents the (causal) dependencies of w on v, and
- for each vertex v with k parents, the function $\Theta_v : D^k \rightarrow Dist(D)$ is the conditional probability table of (the random variable represented by) vertex v.

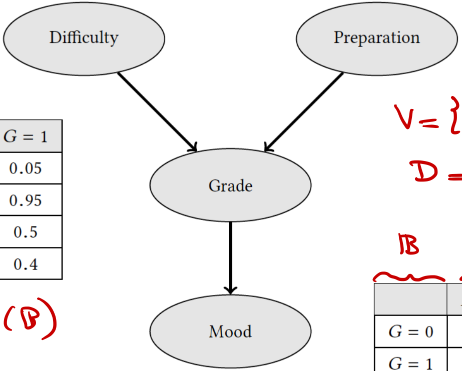
Here, $w \in V$ is a parent of $v \in V$ whenever $(w, v) \in E$.

The graph structure induces a natural ordering on the parents of a vertex v; the i-th entry in a tuple $\mathbf{d} \in D^k$ of Θ_v corresponds to the value assigned to the i-th parent of v.

Example: Student's mood after an exam

$\Theta_D(0) = 0.6$ $\quad \Theta_D : \mathbb{B}^k \to \mathrm{Dist}(\mathbb{B}) \quad k = 0$

$D = 0$	$D = 1$
0.6	0.4



$P = 0$	$P = 1$
0.7	0.3

$V = \{D, P, G, M\}$

$\mathcal{D} = \mathbb{B}$

$\mathbb{B}^2$

	$G = 0$	$G = 1$
$D = 0, P = 0$	0.95	0.05
$D = 1, P = 1$	0.05	0.95
$D = 0, P = 1$	0.5	0.5
$D = 1, P = 0$	0.6	0.4

$\Theta_G : \mathbb{B}^2 \to \mathrm{Dist}(\mathbb{B})$

$\mathbb{B}$ $\quad \mathrm{Dist}(\mathbb{B})$

	$M = 0$	$M = 1$
$G = 0$	0.9	0.1
$G = 1$	0.3	0.7

$\Theta_M : \mathbb{B} \to \mathrm{Dist}(\mathbb{B})$

The interpretation of an entry in a vertex' conditional probability table is:

$$Pr(v = d \mid parents(v) = \mathbf{d}) \;=\; \Theta_v(\mathbf{d})(d), \text{ with } \mathbf{d} \text{ the values of } v\text{'s parents}$$

Bayesian network semantics

$w \in W$
$\wedge\ (v,w) \in E$ implies $v \in W$.

Joint probability function of a Bayesian network

Let $B = (V, E, \Theta)$ be a BN, and $W \subseteq V$ be a downward closed set of vertices where $w \in W$ has value $\underline{w} \in D$. The (unique) joint probability function of BN B in which the nodes in W assume values $\underline{W}$ equals:

$$
\begin{aligned}
Pr(W = \underline{W}) &= \prod_{w \in W} Pr\Big(w = \underline{w} \mid parents(w) = \underline{parents(w)}\Big) \\
&= \underbrace{\prod_{w \in W} \Theta_w(\underline{parents(w)})(\underline{w})}_{\text{also called factorisation}} .
\end{aligned}
$$

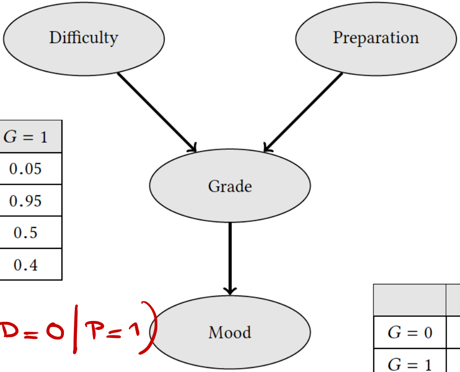
The conditional probability distribution of $W \subseteq V$ given observations on a set $O \subseteq V$ of vertices is given by $Pr(W = \underline{W} \mid O = \underline{O}) = \dfrac{Pr(W = \underline{W} \wedge O = \underline{O})}{Pr(O = \underline{O})}$.

Example

$D = 0$	$D = 1$
0.6	0.4

$P = 0$	$P = 1$
0.7	0.3

	$G = 0$	$G = 1$
$D = 0, P = 0$	0.95	0.05
$D = 1, P = 1$	0.05	0.95
$D = 0, P = 1$	0.5	0.5
$D = 1, P = 0$	0.6	0.4



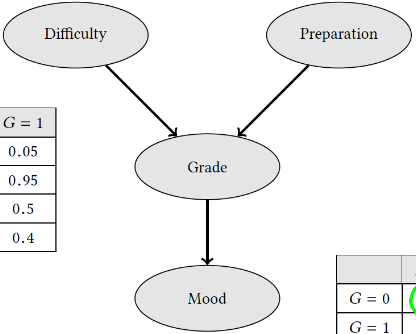
$Pr(M=0, G=0, D=0 \mid P=1)$

	$M = 0$	$M = 1$
$G = 0$	0.9	0.1
$G = 1$	0.3	0.7

How likely does a student end up with a bad mood after getting a bad grade for an easy exam, **given that** she is well prepared?

Example

$D = 0$	$D = 1$
0.6	0.4



$P = 0$	$P = 1$
0.7	0.3

	$G = 0$	$G = 1$
$D = 0, P = 0$	0.95	0.05
$D = 1, P = 1$	0.05	0.95
$D = 0, P = 1$	0.5	0.5
$D = 1, P = 0$	0.6	0.4

	$M = 0$	$M = 1$
$G = 0$	0.9	0.1
$G = 1$	0.3	0.7

$$Pr(D = 0, G = 0, M = 0 \mid P = 1) = \frac{Pr(D = 0, G = 0, M = 0, P = 1)}{Pr(P = 1)}$$

$$= \frac{0.6 \cdot 0.5 \cdot 0.9 \cdot 0.3}{0.3} = \mathbf{0.27}$$

The benefits of Bayesian networks

Bayesian networks provide a compact representation of joint distribution functions

if the dependencies between the random variables are sparse.

Another advantage of BNs is

the explicit representation of conditional independencies.

Overview

Conditional independence

Two independent events may become dependent given some observation. This is captured by the following notion.

Conditional independence

Let X, Y, Z be (discrete) random variables. X is conditionally independent of Y given Z, denoted $I(X, Z, Y)$, whenever:

$$Pr(X \wedge Y \mid Z) = Pr(X \mid Z) \cdot Pr(Y \mid Z) \quad \text{or} \quad Pr(Z) = 0.$$

X and Y are independent

$$Pr(X = x, Y = y) = Pr(X = x) \cdot Pr(Y = y)$$

Conditional independence

Two independent events may become dependent given some observation. This is captured by the following notion.

Conditional independence

Let X, Y, Z be (discrete) random variables. X is conditionally independent of Y given Z, denoted $I(X, Z, Y)$, whenever:

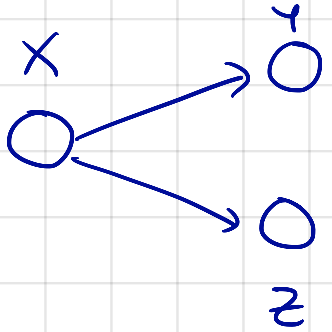
$$Pr(X \wedge Y \mid Z) = Pr(X \mid Z) \cdot Pr(Y \mid Z) \quad \text{or} \quad Pr(Z) = 0.$$

Equivalent formulation: $Pr(X \mid Y \wedge Z) = Pr(X \mid Z)$ or $Pr(Y \wedge Z) = 0$.

These notions can be easily lifted in a point-wise manner to sets of random variables, e.g., $\mathbf{X} = \{ X_1, \ldots, X_k \}$.

Examples on the black board.

a.



<u>Y and Z are not independent</u>

X, Y, Z are unknown

$$\Pr(Y=y, Z=z) = p(y,z)$$

$$p(y,z) = \sum_{x} p(x,y,z)$$ (* marginalisation *)

$$= \sum_{x} p(y|x)\cdot p(z|x)\cdot p(x)$$ (* factorisation *)

$$\neq p(y)\cdot p(z)$$

b. Assume that the value of $X=x$.

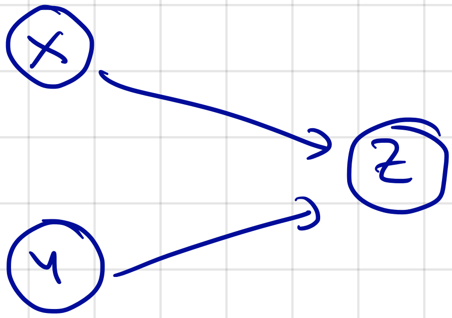
Then: $\Pr(Y=y, Z=z \mid X=x)$

$$p(y,z|x) = \frac{p(x,y,z)}{p(x)}$$ (* cond. prob. *)

$$= \frac{p(y|x)\cdot p(z|x)\cdot \cancel{p(x)}}{\cancel{p(x)}}$$ (* factorisation *)

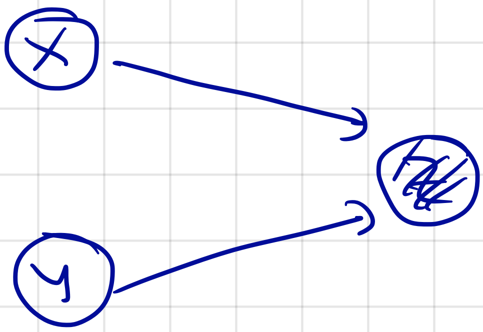
$$= p(y|x)\cdot p(z|x)$$ Thus $I(Y,X,Z)$

e.



X and Y are independent

f.



same but Z is observed / known

$Z = z$

claim: <u>not</u> $I(X, Z, Y)$

c. $X \to Y \to Z$

X and Z are not independent

$$p(x,z) = \sum_y p(x,y,z) = \sum_y p(z|y)\cdot p(y|x)\cdot p(x)$$

$$= p(z|x)\cdot p(x)$$

$$\neq p(z)\cdot p(x)$$

d. $X \to Y \to Z$

assume $Y=y$ is known

$$p(x,z|y) = \frac{p(x,y,z)}{p(y)}$$

$$= \frac{p(x)\cdot p(y|x)\cdot p(z|y)}{p(y)}$$

$$= (* \quad p(x|y) = \frac{p(x,y)}{p(y)} = \frac{p(x)\cdot p(y|x)}{p(y)} \quad *)$$

$$p(x|y)\cdot p(z|y)$$

Graphoid axioms of Bayesian networks

Graphoid axioms **[Dawid, 1979], [Spohn, 1980]**

Conditional independence satisfies the following axioms for disjoint sets of random variables $\mathbf{W}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}$:

1. $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ if and only if $I(\mathbf{Y}, \mathbf{Z}, \mathbf{X})$ — Symmetry
2. $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W})$ implies $\big(I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ and $I(\mathbf{X}, \mathbf{Z}, \mathbf{W})\big)$ — Decomposition
3. $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W})$ implies $I(\mathbf{X}, \mathbf{Z} \cup \mathbf{Y}, \mathbf{W})$ — Weak union
4. $\big(I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ and $I(\mathbf{X}, \mathbf{Z} \cup \mathbf{Y}, \mathbf{W})\big)$ implies $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W})$ — Contraction
5. $I(\mathbf{X}, \mathbf{Z}, \varnothing)$ — Triviality

Decomposition+Weak union+Contraction together are equivalent to:

$$I(\mathbf{X}, \mathbf{Z}, \mathbf{Y} \cup \mathbf{W}) \quad \text{if and only if} \quad I(\mathbf{X}, \mathbf{Z}, \mathbf{Y}) \text{ and } I(\mathbf{X}, \mathbf{Z} \cup \mathbf{Y}, \mathbf{W}).$$

$\Longleftarrow$ contraction

$$I(X, Z, Y \cup W)$$

$\Leftrightarrow$

$$I(X, Z, Y \cup W) \wedge I(X, Z, Y \cup W)$$

$\Rightarrow$ (* decomposition *)

$$I(X, Z, Y) \wedge I(X, Z, W) \wedge I(X, Z, Y \cup W)$$

$\Rightarrow$ (* weak union *)

$$I(X, Z, Y) \wedge I(X, Z, W) \wedge I(X, Z \cup Y, W)$$

$\Rightarrow$

$$I(X, Z, Y) \wedge I(X, Z \cup Y, W) \qquad \boxtimes$$

Checking conditional independencies

Deriving the (conditional) independencies is non-trivial.

The graphical structure of Bayesian networks enable a simple test.

This is based on the concept of d-separation.

Valves

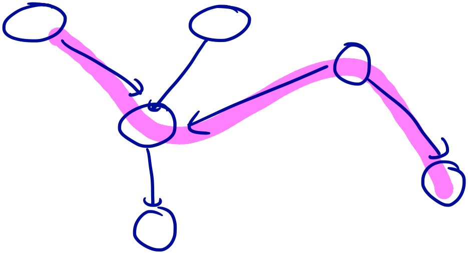
- Consider undirected paths in the underlying DAG $G = (V, E)$ of the BN.

↳ "simple" paths, so no duplicate visits to any vertex

Valves

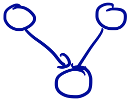
- Consider undirected paths in the underlying DAG $G = (V, E)$ of the BN.

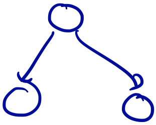
- View every such path as a pipe, and each vertex W on the path as a valve.



Valves

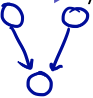
- Consider undirected paths in the underlying DAG $G = (V, E)$ of the BN.
- View every such path as a pipe, and each vertex W on the path as a valve.
- Valves have the status open or closed.
- An undirected path is blocked if at least one valve along the path is closed.
- A valve v is open or closed on a path depending on its type on this path:



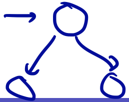


Valves

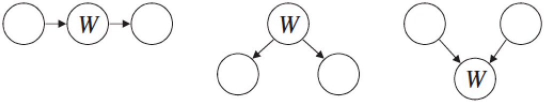
- Consider undirected paths in the underlying DAG $G = (V, E)$ of the BN.
- View every such path as a pipe, and each vertex W on the path as a valve.
- Valves have the status open or closed.
- An undirected path is blocked if at least one valve along the path is closed.



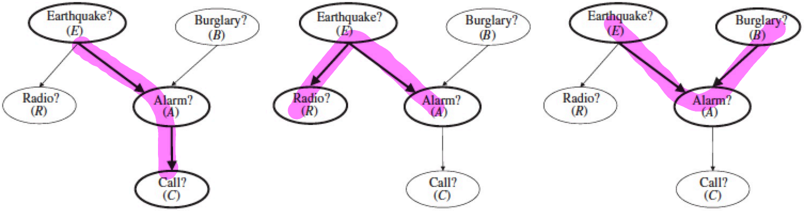
- A valve v is open or closed on a path depending on its type on this path:
  1. Sequential: when v is a parent of one of its neighbours (on the path) and a child of its other neighbour (on the path)
  2. Divergent: when v is a parent of both neighbours
  3. Convergent: when v is a child of both neighbours



Valve types



Sequential Divergent Convergent

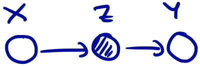


Sequential valve Divergent valve Convergent valve

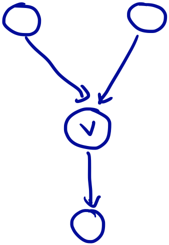
Valve status

set of observed variables

A valve v is closed for set $\mathbf{Z}$ of variables whenever:



1. Sequential: if v (is a variable that) occurs in $\mathbf{Z}$
2. Divergent: if v occurs in $\mathbf{Z}$
3. Convergent: if neither v nor any of its descendants occurs in $\mathbf{Z}$.
   w is a descendant of v if w is reachable via (directed) edge relation E from v.



Valve status

A valve v is closed for set $\mathbf{Z}$ of variables whenever:

1. Sequential: if v (is a variable that) occurs in $\mathbf{Z}$
2. Divergent: if v occurs in $\mathbf{Z}$
3. Convergent: if neither v nor any of its descendants occurs in $\mathbf{Z}$.
   w is a descendant of v if w is reachable via (directed) edge relation E from v.

Example

1. the sequential valve A is closed iff we know the value of A, otherwise an earthquake E may change our belief in getting a call C.
2. the divergent valve E is closed iff we know the value of variable E, otherwise a radio report on an earthquake may change our belief in the alarm triggering.
3. the convergent valve A is closed iff neither the value of variable A nor the value of C are known, otherwise, a burglary may change our belief in an earthquake.

D-separation

D-separation

Let **X**, **Y**, **Z** be disjoint sets of vertices in the DAG G. **X** and **Y** are d-separated by **Z** in G, denoted $dsep_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$, iff every (undirected) path between a vertex in **X** and a vertex in **Y** is blocked by some vertex in **Z**.

A path is blocked by **Z** iff at least one vertex on the path is closed given **Z**.

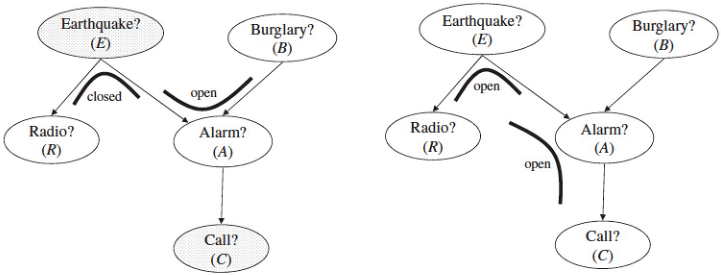


Figure 4.9: On the left, R and B are d-separated by E, C. On the right, R and C are not d-separated.

D-separation

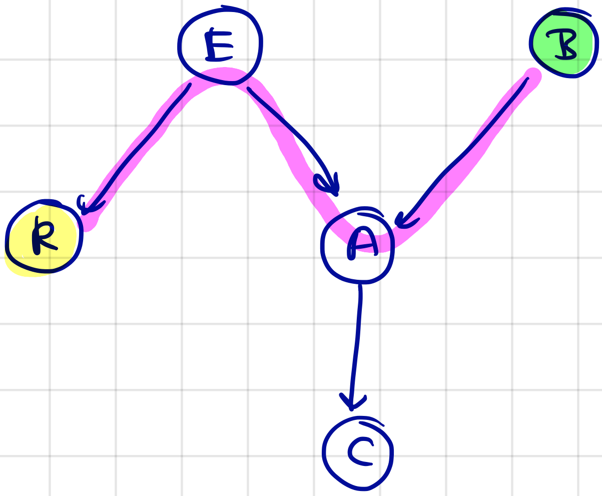
D-separation implies independence [Pearl 1986], [Verma, 1986]

$dsep_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ implies $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$.

Proof.

Left as an exercise. Note that the reverse implication does not hold. □

G:

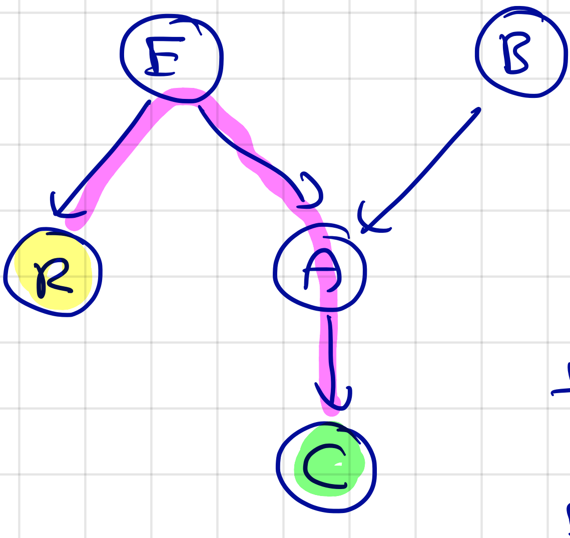


$dsep_G(R, \underbrace{\{E, C\}}, B)$

- there is only one path between R and B.
- two valves: A and E
  - E is closed. E is divergent and $E \in \{E, C\}$

$\Rightarrow$ path is blocked $\quad I(R, \{E, C\}, B)$

2



not $dsep(R, \emptyset, C)$

two valves: E, A

E is divergent $\quad E \notin \emptyset$

A is sequential $\quad A \notin \emptyset$

$\Rightarrow$ E, A are open

D-separation

D-separation implies independence **[Pearl 1986], [Verma, 1986]**

$dsep_G(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$ implies $I(\mathbf{X}, \mathbf{Z}, \mathbf{Y})$.

Proof.

Left as an exercise. Note that the reverse implication does not hold. □

As d-separation is defined over all paths, this theorem yields an **exponential-time procedure to check (a sufficient condition for) conditional independence.**

A polynomial algorithm for d-separation

Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ be disjoint sets of vertices in the DAG G. Apply the following pruning procedure on the DAG G:

1. Eliminate any leaf vertex v from G with $v \notin (\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z})$.
2. Repeat this elimination procedure until no more leafs can be eliminated.
3. Eliminate all edges emanating from vertices in $\mathbf{Z}$.

The remaining DAG is referred to as $\mathit{prune}_{\mathbf{X},\mathbf{Y},\mathbf{Z}}(G)$.

A polynomial algorithm for d-separation

Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ be disjoint sets of vertices in the DAG G. Apply the following pruning procedure on the DAG G:

1. Eliminate any leaf vertex v from G with $v \notin (\mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z})$.
2. Repeat this elimination procedure until no more leafs can be eliminated.
3. Eliminate all edges emanating from vertices in $\mathbf{Z}$.

a.k.a: outgoing

The remaining DAG is referred to as $prune_{\mathbf{X},\mathbf{Y},\mathbf{Z}}(G)$.

Theorem

Let $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ be disjoint sets of vertices in the DAG G. Then:

$$dsep_G(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \text{ iff } \mathbf{X} \text{ and } \mathbf{Y} \text{ are disconnected in } prune_{\mathbf{X},\mathbf{Y},\mathbf{Z}}(G).$$

Two sets of vertices are disconnected if there is no path between them.

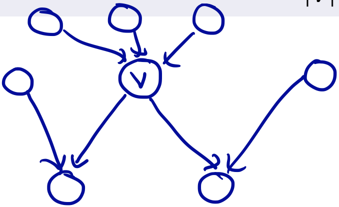
Markov blanket

The complexity of inference on a Bayesian network is measured in terms of the Markov blanket, an indication of the degree of dependence in the BN.

Markov blanket

The Markov blanket for a vertex v in a BN is the set ∂v of vertices composed of v, v's parents, its children, and its children's other parents.

The average Markov blanket of BN B is the average size of the Markov blanket of all its vertices, that is, $\frac{1}{|V|}\sum_{v \in V} |\partial v|$.



Markov blanket

The complexity of inference on a Bayesian network is measured in terms of the Markov blanket, an indication of the degree of dependence in the BN.

Markov blanket

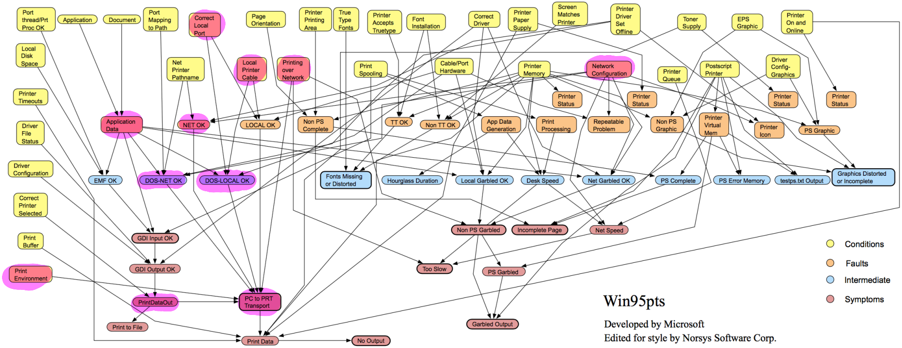
The Markov blanket for a vertex v in a BN is the set ∂v of vertices composed of v, v's parents, its children, and its children's other parents.

The average Markov blanket of BN B is the average size of the Markov blanket of all its vertices, that is, $\frac{1}{|V|} \sum_{v \in V} |\partial v|$.

Every set of nodes in the BN is conditionally independent of v when conditioned on the set ∂v. That is, for distinct vertices v and w:

$$Pr(v \mid \partial v \wedge w) = Pr(v \mid \partial v) \quad \text{or, equivalently} \quad I(\{ v \}, \partial v, \{ w \})$$

Printer troubleshooting in Windows 95



The average Markov blanket of this BN is 5.92, $|V| = 76$, and $|E| = 117$

= Markov blanket of vertex "DOS-LOCAL OK".

Some benchmark BN results

Benchmark BNs from `www.bnlearn.com`

BN	$\|V\|$	$\|E\|$	**aMB**
`hailfinder`	56	66	3.54
`hepar2`	70	123	4.51
`win95pts`	76	112	5.92
`pathfinder`	135	200	3.04
`andes`	223	338	5.61
`pigs`	441	592	3.92
`munin`	1041	1397	3.54

aMB = *average Markov Blanket size*, a measure of independence in BNs

Overview

Probabilistic inference

We consider the following probabilistic inference problem: let B be a BN with set V of vertices and the evidence $\mathbf{E} \subseteq V$ and the questions $\mathbf{Q} \subseteq V$. (Exact) probabilistic inference is to determine the conditional probability

$$Pr(\mathbf{Q} = \mathbf{q} \mid \mathbf{E} = \mathbf{e}) = \frac{Pr(\mathbf{Q} = \mathbf{q} \wedge \mathbf{E} = \mathbf{e})}{Pr(\mathbf{E} = \mathbf{e})}.$$

We consider:

Decision variants of probabilistic inference

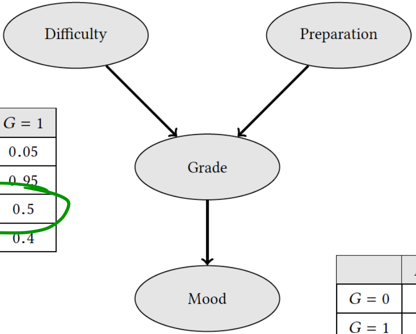
The decision variant of probabilistic inference is: for a given probability $p \in \mathbb{Q} \cap [0, 1)$:

- does $Pr(\mathbf{Q} = \mathbf{q} \mid \mathbf{E} = \mathbf{e}) > p$? TI[1]
- special case: $Pr(\mathbf{E} = \mathbf{e}) > p$? STI

[1] TI = Threshold Inference and STI = Simple TI.

Example

$D = 0$	$D = 1$
0.6	0.4



$P = 0$	$P = 1$
0.7	0.3

	$G = 0$	$G = 1$
$D = 0, P = 0$	0.95	0.05
$D = 1, P = 1$	0.05	0.95
$D = 0, P = 1$	0.5	0.5
$D = 1, P = 0$	0.6	0.4

	$M = 0$	$M = 1$
$G = 0$	0.9	0.1
$G = 1$	0.3	0.7

$$Pr(D = 0, G = 0, M = 0 \mid P = 1) = \frac{Pr(D = 0, G = 0, M = 0, P = 1)}{Pr(P = 1)}$$

$$= \frac{0.6 \cdot 0.5 \cdot 0.9 \cdot 0.3}{0.3} = \mathbf{0.27}$$

Complexity of probabilistic inference

Decision variants of probabilistic inference

For a given probability $p \in \mathbb{Q} \cap [0, 1)$:

- does $Pr(\mathbf{Q} = \mathbf{q} \mid \mathbf{E} = \mathbf{e}) \; > \; p$? TI
- special case: $Pr(\mathbf{E} = \mathbf{e}) \; > \; p$? STI

Complexity of probabilistic inference [Cooper, 1990]

The decision problems TI and STI are PP-complete.

Proof.

1. Hardness: by a reduction of MAJSAT to STI (since STI is a special case of TI, MAJSAT is reducible to TI).
2. Membership: To show TI is in PP, a polynomial-time algorithm is provided that can guess a solution to TI while guaranteeing that the guess is correct with probability exceeding $1/2$.

□

The complexity class PP

PP (Probabilistic Polynomial-Time) is the class of decision problems solvable by a probabilistic Turing machine[2] in polynomial time with an error probability $< 1/2$.

Formally, a language L is in PP iff there is a probabilistic TM M such that:

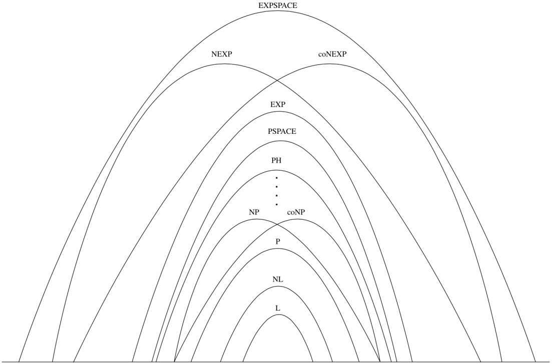
1. M runs in polynomial time on all inputs
2. For all $w \in L$, M outputs 1 with probability larger than $1/2$
3. For all $w \notin L$, M outputs 1 with probability at most $1/2$.

A PP-problem can be solved to any fixed degree of accuracy by running a randomised polynomial-time algorithm a sufficient (but bounded) number of times.

Remark: if all choices are binary and the probability of each transition is $1/2$, then the majority of the runs accept input w iff $w \in L$. This majority, however, is not fixed and may (exponentially) depend on the input, e.g., a problem in PP may accept "yes"-instances with size $|w|$ with probability $1/2 + \frac{1}{2^{|w|}}$. This makes problems in PP intractable in general.

[2] A probabilistic TM is a non-deterministic TM which chooses between the available transitions at each point according to some probability distribution.

The complexity class PP



NP ⊆ PP (as SAT lies in PP) and coNP ⊆ PP (as PP is closed under complement). PP is contained in PSPACE (as there is a polynomial-space algorithm for MAJSAT).

PP is comparable to the class #P — the counting variant of NP — the class of function problems "compute $f(x)$" where f is the number of accepting runs of an NTM running in polynomial time.

The decision problems SAT and MAJSAT

The decision problems SAT and MAJSAT

Let α be a propositional logical formula (in conjunctive normal form, CNF) over a finite set $\mathbf{X}$ of Boolean variables.

1. Does there exist a valuation over $\mathbf{X}$ such that α holds? SAT
2. Does the majority of the assignments to $\mathbf{X}$ make α hold? MAJSAT

Known facts [Cook, 1971] and [??]

1. The SAT problem is NP-complete.
2. The MAJSAT problem is PP-complete.

Showing membership

By providing a polynomial-time algorithm that can guess a solution to TI while guaranteeing that the guess is correct with probability exceeding $1/2$.

Showing hardness

threshold inference

Aim: show that STI / TI is PP-hard.

strategy: reduce a $\underbrace{\text{known PP-hard problem}}_{\text{MAJSAT.}}$ to STI

CNF sentence α over $\{X_1, \ldots, X_n\}$ $\longmapsto$ BN B_α (with leaf V_α)

s.t.

$\underbrace{\text{MAJSAT}(\alpha)}$ iff $Pr(V_\alpha = \text{true}) > \frac{1}{2}$

i.e. the majority of the 2^n possible assignments to $\{X_1, \ldots, X_n\}$ satisfy α

$\downarrow$

all vertices in B_α are binary

How to obtain BN_α from sentence α?

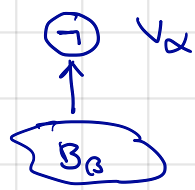
Inductively over the structure of α. $D = \mathbb{B}$

① $\alpha = X_i$ then BN_α has $V = \{X_i\}$ $\Theta_V(\text{true}) = \frac{1}{2}$

$\Theta_V(\text{false}) = \frac{1}{2}$

② $\alpha = \neg\beta$ then $V = \{V_\alpha\} \cup$ vertices in BN_β

$E = \{(V_\alpha, \text{root } B_\beta)\}$

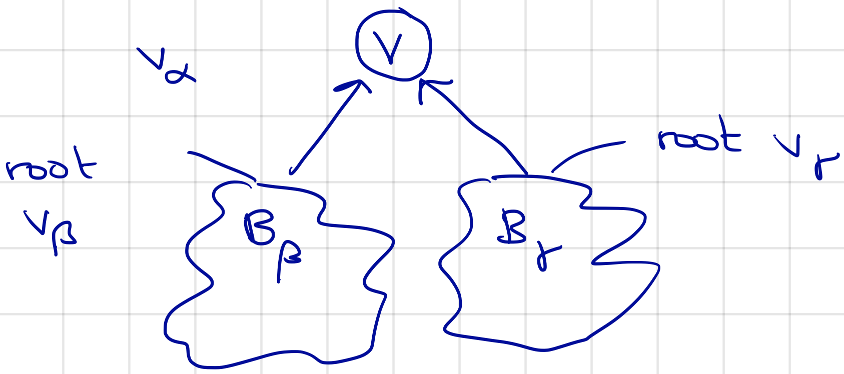


cpt for vertex v_α:

$\Theta(v_\alpha)$:

root B_β	v_α	$\Theta(v_\alpha)$
tt	tt	0
ff	tt	1

(3) $\alpha = \beta \vee \gamma$

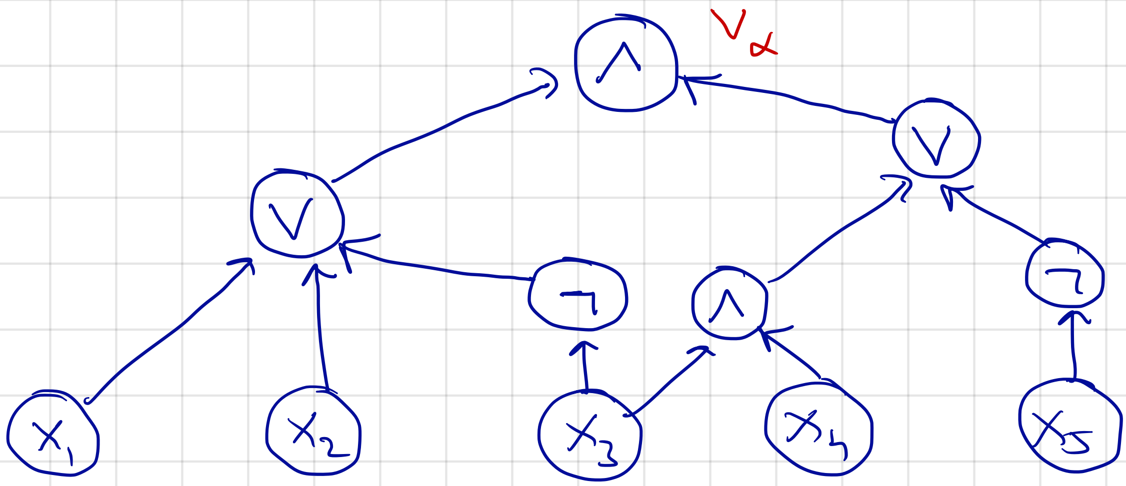


cpt for v_α

root v_β	root v_γ	v_α	$\Theta(v_\alpha)$
tt	tt	tt	1
tt	ff	tt	1
ff	tt	tt	1
ff	ff	tt	0

(4) $\wedge$: along similar lines.

Ex: $\alpha = (x_1 \vee x_2 \vee \neg x_3) \wedge ((x_3 \wedge x_4) \vee \neg x_5)$



Lemma:

$$\Pr(x_1, \ldots, x_n, V_\alpha = tt) = \begin{cases} 0 & \text{if } x_1, \ldots, x_n \not\models \alpha \\ \frac{1}{2^n} & \text{if } x_1, \ldots, x_n \models \alpha \end{cases}$$

(left as exercise)

Theorem $\text{MAJSAT}(x_1, \ldots, x_n \models \alpha)$ iff $\Pr(V_\alpha = tt) > \frac{1}{2}$

Proof: $\Pr(V_\alpha = tt)$

$$= \sum_{x_1, \ldots, x_n} \Pr(V_\alpha = tt, x_1, \ldots, x_n)$$

(* jdf of BN B_α *)

$$= \sum_{x_1, \ldots, x_n \models \alpha} \Pr(V_\alpha = tt, x_1, \ldots, x_n) + \underbrace{\sum_{x_1, \ldots, x_n \not\models \alpha} \Pr(V_\alpha = tt, x_1, \ldots, x_n)}_{= 0}$$

(* above lemma *)

$$= \frac{1}{2^n} \cdot \underset{\uparrow}{c} + 0$$

$\#$ instantiations of $x_1, \ldots, x_n \models \alpha$

$$= \frac{c}{2^n}$$

α has 2^n possible instantiations

majority of these instantiations $\models \alpha$ when $c > \frac{2^n}{2}$

$c > \frac{2^n}{2}$ iff $\frac{c}{2^n} > \frac{1}{2}$ iff (see above) $\Pr(V_\alpha = tt) > \frac{1}{2}$ $\boxtimes$.

Showing membership $\Pr(Q=q \mid E=e) > p$

Aim: show that Ti $\in$ PP.

How: give a polytime algorithm for Ti that is correct with probability $> \frac{1}{2}$

1. let $a(p) = \begin{cases} 1 & \text{if } p < \frac{1}{2} \\ \frac{1}{2p} & \text{else} \end{cases}$ $\qquad b(p) = \begin{cases} \frac{(1-2p)}{2-2p} & \text{if } p < \frac{1}{2} \\ 0 & \text{else} \end{cases}$

2. sample an instantiation $\underline{x}$ for $X_1, \dots, X_n$ randomly

3. declare $P(q \mid e) > p$ with probability

$$\begin{cases} a(p) & \text{if } \underline{x} \text{ is compatible with } q \text{ and } e \\ b(p) & \text{" " " with } e\text{, but not } q \\ \frac{1}{2} & \text{" " " not compatible with } e. \end{cases}$$

Theorem This algorithm declares $P(q \mid e) > p$ with prob. $> \frac{1}{2}$

Proof: the prob. of declaring $P(q \mid e) > p$ is given by:

$$r = a(p) \cdot \Pr(q, e) + b(p) \cdot \Pr(\neg q, e) + \underbrace{\tfrac{1}{2}\big(1 - \Pr(e)\big)}_{\frac{1}{2} - \frac{1}{2}\Pr(e)}$$

Thus $\boxed{r > \frac{1}{2} \text{ iff } a(p) \cdot \Pr(q,e) + b(p) \cdot \Pr(\neg q, e) > \frac{1}{2}\Pr(e)}$

$$\Leftrightarrow \quad a(p) \cdot \Pr(q \mid e) + b(p) \cdot \Pr(\neg q \mid e) > \frac{1}{2}$$

$$r > \frac{1}{2} \quad \text{iff} \quad a(p) \cdot Pr(g|e) + b(p) \cdot Pr(\neg g|e) > \frac{1}{2}$$

Distinguish two cases:

1. $p < \frac{1}{2}$. Then: $a(p) \cdot Pr(g|e) + b(p) \cdot Pr(\neg g|e) > \frac{1}{2}$

$$\Leftrightarrow \quad 1 \cdot Pr(g|e) + \left(\frac{1-2p}{2-2p}\right) \cdot \left(1 - Pr(g|e)\right) > \frac{1}{2}$$

$$\Leftrightarrow \quad Pr(g|e)\left(1 - \frac{1-2p}{2-2p}\right) > \frac{1}{2} - \frac{1-2p}{2-2p}$$

$$\Leftrightarrow \quad Pr(g|e)\left(\frac{1}{2-2p}\right) > \frac{p}{2-2p}$$

$$\Leftrightarrow \quad Pr(g|e) > p$$

2. $p > \frac{1}{2}$ Then: $a(p) \cdot Pr(g|e) + b(p) \cdot Pr(\neg g|e) > \frac{1}{2}$

$$\Leftrightarrow \quad \frac{1}{2p} Pr(g|e) + 0 > \frac{1}{2}$$

$$\Leftrightarrow \quad Pr(g|e) > p.$$

Thus $r > \frac{1}{2}$ iff $Pr(g|e) > p$ $\boxtimes$.