Bayesian AI
Tutorial

Ann E. Nicholson and **Kevin B. Korb**

Faculty of Information Technology
Monash University
Clayton, VIC 3168
AUSTRALIA

{annn,korb}@csse.monash.edu.au

HTTP://WWW.CSSE.MONASH.EDU.AU/BAI

Text: *Bayesian Artificial Intelligence*, Kevin B. Korb and Ann E. Nicholson, Chapman & Hall/CRC, 2004.

Schedule

9.30 Welcome

9.35 Bayesian AI

Introduction to Bayesian networks

Reasoning with Bayesian networks

11.00 Morning Tea break

11.15 Decision networks

Dynamic Bayesian networks

1.15 Lab session (Netica, Matilda)

2.30 Afternoon Tea break

2.45 Learning Bayesian networks

Knowledge Engineering with Bayesian networks (KEBN)

KEBN software (CaMML, VerbalBN)

4.00 FINISH

Introduction to Bayesian AI

- Reasoning under uncertainty
- Probabilities
- Bayesian philosophy
  - Bayes' Theorem
  - Conditionalization
  - Motivation
- Bayesian decision theory
- How to be an effective Bayesian
- Probabilistic causality
  - Humean causality
  - Prob causality
  - Are Bayesian networks Bayesian?
- Towards a Bayesian AI

Reasoning under uncertainty

Uncertainty: The quality or state of being not clearly known.

This encompasses most of what we understand about the world — and most of what we would like our AI systems to understand.

> Distinguishes *deductive* knowledge (e.g., mathematics) from *inductive* belief (e.g., science).

Sources of uncertainty

- Ignorance
  (which side of this coin is up?)
- Complexity
  (meteorology)
- Physical randomness
  (which side of this coin will land up?)
- Vagueness
  (which tribe am I closest to genetically? Picts? Angles? Saxons? Celts?)

Probability Calculus

Classic approach to reasoning under uncertainty.
(origin: Blaise Pascal and Fermat).

Kolmogorov's Axioms:

1. $P(U) = 1$
2. $\forall X \subseteq U \; P(X) \geq 0$
3. $\forall X, Y \subseteq U$
   if $X \cap Y = \emptyset$
   then $P(X \vee Y) = P(X) + P(Y)$

Conditional Probability $P(X|Y) = \frac{P(X \wedge Y)}{P(Y)}$

Independence $X \perp\!\!\!\perp Y$ iff $P(X|Y) = P(X)$

Probability Theory

So, why not use probability theory to represent uncertainty?
That's what it was invented for... dealing with physical randomness and degrees of ignorance.

Justifications:

- Ramsey (1926): Derives axioms from maximization of expected utility
  - Dutch books
- Cox (1946): Derives isomorphism to probability two simpler axioms –
  - $B(\neg\Phi)$ is a function of $B(\Phi)$
  - $B(\Phi \wedge \Phi')$ is a fnc of $B(\Phi|\Phi')$ and $B(\Phi')$

Probability theory for representing uncertainty

- Propositions are either true OR false
  - e.g. “it will rain today”
- Assign a numerical degree of belief between 0 and 1 to facts (propositions)
  - e.g. P(“it will rain today”) = 0.2
  - This is a *prior probability* (unconditional)
- Given other information, can express conditional beliefs
  - e.g. P(“it will rain today” | “rain is forecast”) = 0.8
  - Called *posterior* or *conditional* probability

Rev. Thomas Bayes
(1702-1761)

Bayes' Theorem; Conditionalization

— Due to Reverend Thomas Bayes (1764)

$$P(h|e) = \frac{P(e|h)P(h)}{P(e)}$$

Conditionalization: $P'(h) = P(h|e)$

Or, read Bayes' theorem as:

$$\text{Posterior} \ = \ \frac{\text{Likelihood} \ \times \ \text{Prior}}{\text{Prob of evidence}}$$

Assumptions:

1. Joint priors over $\{h_i\}$ and e exist.
2. Total evidence: e, and only e, is learned.

Motivation: Breast Cancer

Let $P(h) = 0.01$ (one in 100 women tested have it)
$P(e|h) = 0.8$ and $P(e|\neg h) = 0.1$
(true and false positive rates). What is $P(h|e)$?

Motivation: Breast Cancer

Let $P(h) = 0.01$ (one in 100 women tested have it)
$P(e|h) = 0.8$ and $P(e|\neg h) = 0.1$
(true and false positive rates). What is $P(h|e)$?

Bayes' Theorem yields:

$$\begin{aligned}
P(h|e) &= \frac{P(e|h)P(h)}{P(e)} \\
&= \frac{P(e|h)P(h)}{P(e|h)P(h) + P(e|\neg h)P(\neg h)} \\
&= \frac{0.8 \times 0.01}{0.8 \times 0.01 + 0.1 \times 0.99} \\
&= \frac{0.008}{0.008 + 0.099} \\
&= \frac{0.008}{0.107} \\
&\approx 0.075
\end{aligned}$$

Will there be a hail storm?

$P(hailStorm) =$

What sort of evidence, E, might become available?

$P(E|hailStorm) =$

$P(E|\neg hailStorm) =$

$P(hailStorm|E) =$

Motivation

Huge variety of cases where

- Uncertainty dominates considerations
- Getting it right is crucial

Examples and consequences:

- Weather forecasting:
  - Fog: aviation industry – safety, costs
  - Hailstorm: emergencies services, insurance costs
  - ...
- Medicine: death, injury, disease
- Law: false imprisonment, wrongful execution
- Space shuttle: explosion
- Hiring: wasted time and money

Bayesian Decision Theory

— Frank Ramsey (1926)

Decision making under uncertainty: what action to take (plan to adopt) when future state of the world is not known.

Bayesian answer: Find utility of each possible outcome (action-state pair) and take the action that **maximizes expected utility**.

Example

Action	Rain (p = .4)	Shine (1 - p = .6)
Take umbrella	30	10
Leave umbrella	-100	50

Expected utilities:

E(Take umbrella) = (30)(.4) + (10)(.6) = 18

E(Leave umbrella) = (-100)(.4) + (50)(.6) = -10

Decision theory: fog example

Action	Fog (p =)	No Fog (1 - p =)
Forecast Fog		
Forecast No Fog		

NB: Full set of Forecast actions: {NoFog, CodeGrey¡5%chance, CodeGrey5%chance, CodeGrey10%chance, CodeGrey20%chance, TAFProbablyFog}

Expected utilities:
E(Forecast Fog) =

E(Forecast No Fog) =

How to be an effective Bayesian

Evidence shows human intuition is *ineffective*. (e.g., Kahneman et al., 1982)

How to get it right?

- Stop relying on intuition:
  - Use paper & pencil, calculator, etc to apply Bayes' thm properly
  - Use frequency formats (Gigerenzer & Hoffrage, 1995)
  - Use Bayesian networks (e.g., Netica)

Humean Causality

As we shall see, causality and Bayesian networks are intimately related concepts.

David Hume (1737) analyzed A *causes* B as:

- Whenever A occurs, B occurs
- A and B are contiguous
- A precedes B

This was immediately challenged by Thomas Reid: let A be night and B day; the conditions are satisfied, but neither causes the other.

> Leading to: CounterExample $\rightarrow$ new conditions $\rightarrow$ CE $\ldots$

Through the next two centuries the "conditions" (sufficiency) account of causality has built up complexity without explanation

Probabilistic Causality

Salmon (1980): What is this sufficiency nonsense? Either of determinism and indeterminism are *possible* – i.e., it is a contingent fact of the world whether it is deterministic or not.

1. A philosophical analysis of causality should not *presume* determinism.
2. Besides, there is evidence that indeterminism is correct.
3. A probabilistic analysis of causality does predetermine the determinism question, whereas the sufficiency analysis does.

Probabilistic causality

- started by H Reichenbach, IJ Good, P Suppes, W Salmon
- has turned into the theory of Bayesian networks

Are Bayesian networks Bayesian?

Bayesianism is a thesis about probabilities: that they represent (ideal) subjective degrees of belief.

If *Bayesian nets* represent probabilistic causal relations, in what way are they subjective?

Lewis's Principal Principle: $P(h|Ch(h) = r) = r$

> If we learn that A probabilistically causes B with chance r under the known circumstances, we are irrational if we don't attribute r to the chance of A causing B.

In other words, Bayesian probability subsumes physical probability.

Bayesian AI

A Bayesian conception of an AI is:
An autonomous agent which

- Has a utility structure (preferences)
- Can learn about its world and the relation between its actions and future states (probabilities)
- Maximizes its expected utility

The techniques used in learning about the world are (primarily) statistical… Hence

Bayesian data mining

Introduction to Bayesian Networks

- Nodes, structure and probabilities
- Reasoning with BNs
- Understanding BNs

Bayesian Networks

- Data Structure which represents the dependence between variables.
- Gives concise specification of the joint probability distribution.
- A Bayesian Network is a graph in which the following holds:

  1. A set of random variables makes up the nodes in the network.
  2. A set of directed links or arrows connects pairs of nodes.
  3. Each node has a conditional probability table that *quantifies* the effects the parents have on the node.
  4. Directed, acyclic graph (DAG), i.e. no directed cycles.

Example: Lung Cancer Diagnosis

A patient has been suffering from shortness of breath (called dyspnoea*) and visits the doctor, worried that he has lung cancer. The doctor knows that other diseases, such as tuberculosis and bronchitis are possible causes, as well as lung cancer. She also knows that other relevant information includes whether or not the patient is a smoker (increasing the chances of cancer and bronchitis) and what sort of air pollution he has been exposed to. A positive XRay would indicate either TB or lung cancer.*

Nodes and Values

Q: What are the nodes to represent and what values can they take?

Nodes can be discrete or continuous; will focus on discrete for now.

- Boolean nodes: represent propositions, taking binary values true (T) and false (F).

  Example: *Cancer* node represents proposition "the patient has cancer".

- Ordered values..

  Example: *Pollution* node with values $\{$*low, medium, high*$\}$.

- Integral values.

  Example: *Age* node with possible values from 1 to 120.

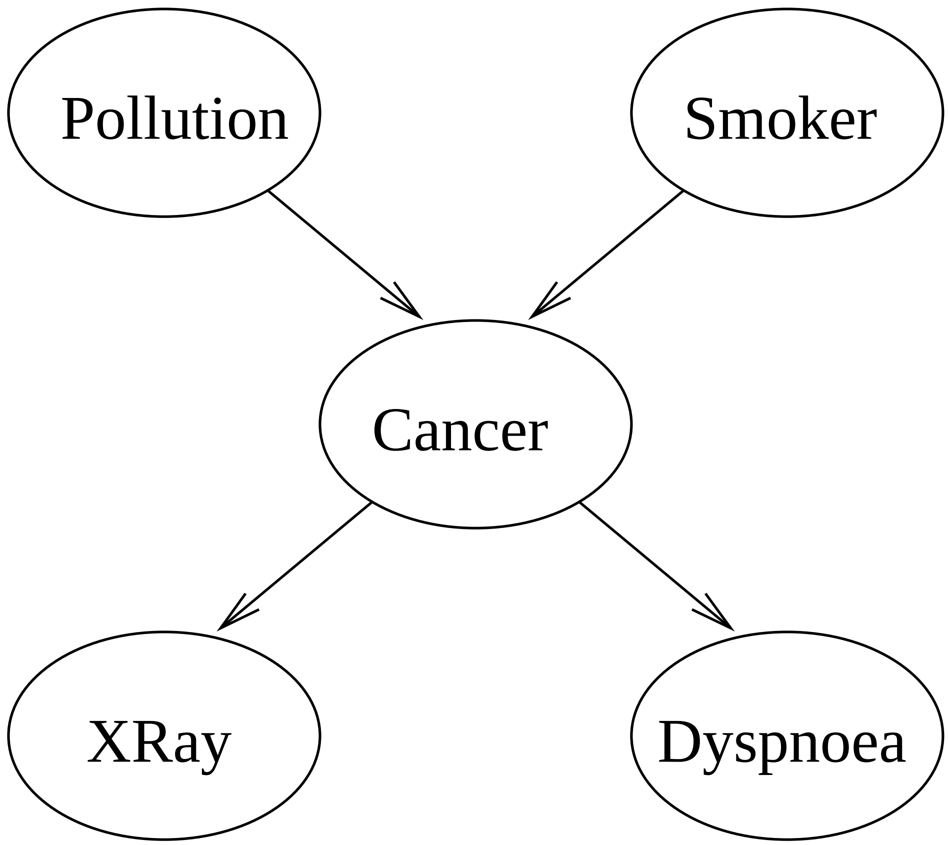
Nodes and Values (cont.)

Weather forecasting examples?

Lung cancer example: nodes and values

Node name	Type	Values
Pollution	Binary	$\{low, high\}$
Smoker	Boolean	$\{T, F\}$
Cancer	Boolean	$\{T, F\}$
Dyspnoea	Boolean	$\{T, F\}$
XRay	Binary	$\{pos, neg\}$

Lung cancer example: network structure



Note: No explicit representation of other causes of cancer, or other causes of symptoms.

Structure terminology and layout

- Family metaphor:

  Parent $\Rightarrow$ *Child*

  Ancestor $\Rightarrow \ldots \Rightarrow$ *Descendant*

- Markov Blanket = parents + children + children's parents

- Tree analogy:
  - **root** node: no parents
  - **leaf** node: no children
  - **intermediate** node: non-leaf, non-root

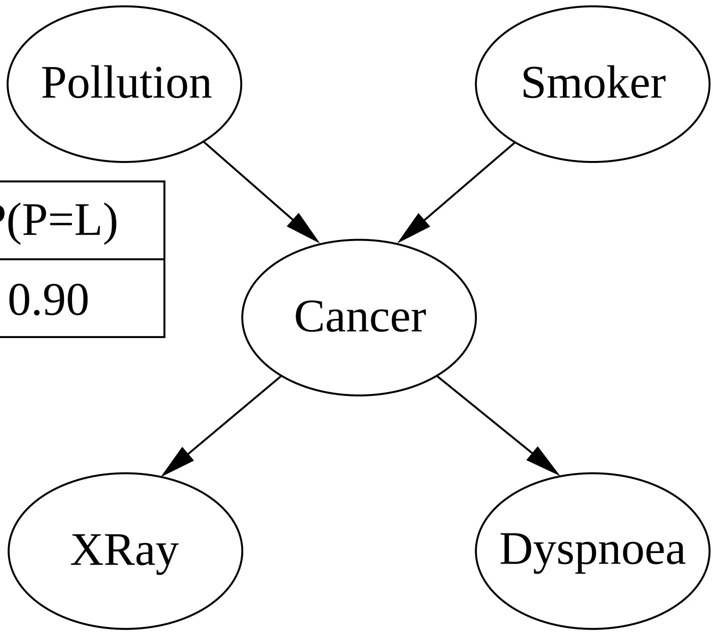
- Layout convention: root notes at top, leaf nodes at bottom, arcs point down the page.

Conditional Probability Tables

Once specified topology, need to specify **conditional probability table** (CPT) for each node.

- Each row contains the conditional probability of each node value for a each possible combination of values of its parent nodes.
- Each row must sum to 1.
- A table for a Boolean var with n Boolean parents contain 2^{n+1} probs.
- A node with no parents has one row (the prior probabilities)

Lung cancer example: CPTs



P(P=L)
0.90

P(S=T)
0.30

P	S	P(C=T|P,S)
H	T	0.05
H	F	0.02
L	T	0.03
L	F	0.001

C	P(X=pos|C)
T	0.90
F	0.20

C	P(D=T|C)
T	0.65
F	0.30

The Markov Property

- Modelling with BNs requires the assumption of the **Markov Property**:

  there are no direct dependencies in the system being modeled which are not already explicitly shown via arcs.

- Example: there is no way for smoking to influence dyspnoea except by way of causing cancer.

- BNs which have the Markov propertly are called Independence-Maps (I-Maps).

- Note: existence of arc does not have to correspond to real dependency in the system being modelled - can be nullified in the CPT.

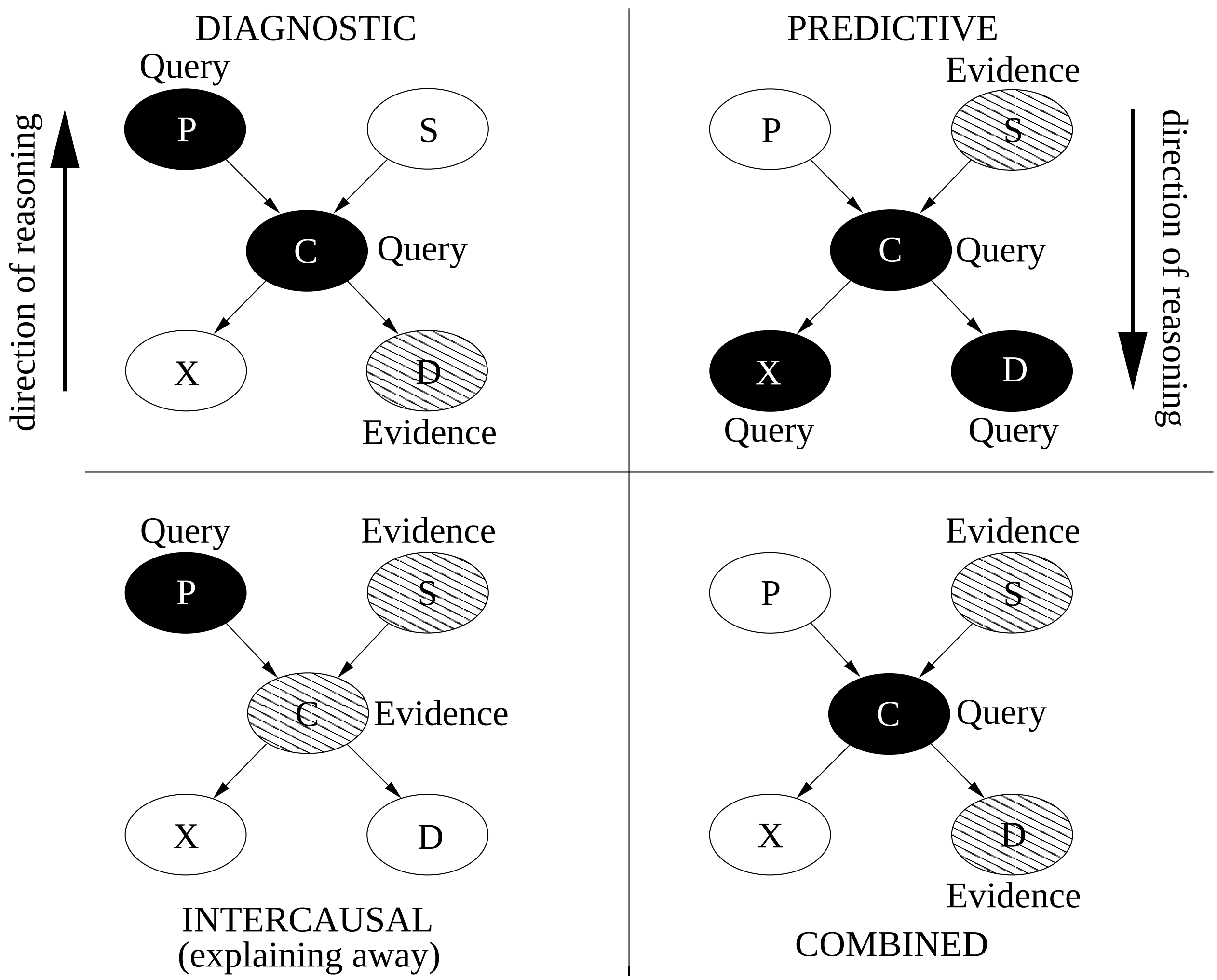
Reasoning with Bayesian Networks

- Basic task for any probabilistic inference system:

  Compute the posterior probability distribution for a set of **query variables**, given new information about some **evidence variables**.

- Also called *conditioning* or *belief updating* or *inference*.

Types of Reasoning



Types of Evidence

- Specific evidence: a definite finding that a node X has a particular value, x.

  Example: *Smoker*=*T*

- Negative evidence: a finding that node Y is *not* in state y_1 (but may take any other values).

- "Virtual" or "likelihood" evidence: source of information is not sure about it.

  Example:

  - $e =$ Radiologist is 80% sure that *Xray*=*pos*
  - Want e.g.:

  $$P(Cancer|e) = P(Cancer|Xray, e)P(Xray|e) + P(Cancer|\neg Xray, e)P(\neg Xray|e)$$

  - **Jeffrey Conditionalization**

Reasoning with numbers

- Reasoning with lung cancer example using Netica BN software.

(See Table 2.2 in *Bayesian AI* text.)

Understanding of Bayesian Networks (Semantics)

- A (more compact) representation of the joint probability distribution.
  - helpful in understanding how to construct network
- Encoding a collection of conditional independence statements.
  - helpful in understanding how to design inference procedures
  - via *Markov property/I-map:*

    Each conditional independence implied by the graph is present in the probability distribution

Representing the joint probability distribution

Write $P(X_1 = x_1, \ldots, X_n = x_n)$ as $P(x_1, x_2, \ldots, x_n)$.

Factorization (chain rule):

$$\begin{aligned} P(x_1, x_2, \ldots, x_n) &= P(x_1) \times \ldots \times P(x_n | x_1, \ldots, x_{n-1}) \\ &= \prod_i P(x_i | x_1, \ldots, x_{i-1}) \end{aligned}$$

Since BN structure implies that the value of a particular node is conditional *only* on the values of its parent nodes, this reduces to

$$P(x_1, x_2, \ldots, x_n) = \prod_i P(x_i | Parents(X_i))$$

provided $Parents(X_i) \subseteq \{x_1, \ldots, x_{i-1}\}$.

$P(X = pos \wedge D = T \wedge C = T \wedge P = low \wedge S = F)$

$$\begin{aligned} &= P(X = pos | D = T, C = T, P = lo, S = F) \\ &\quad \times P(D = T | C = T, P = lo, S = F) \\ &\quad \times P(C = T | P = lo, S = F) P(P = lo | S = F) P(S = F) \\ &= P(X = pos | C = T) P(D = T | C = T) \\ &\quad \times P(C = T | P = lo, S = F) P(P = lo) P(S = F) \end{aligned}$$

Pearl's Network Construction Algorithm

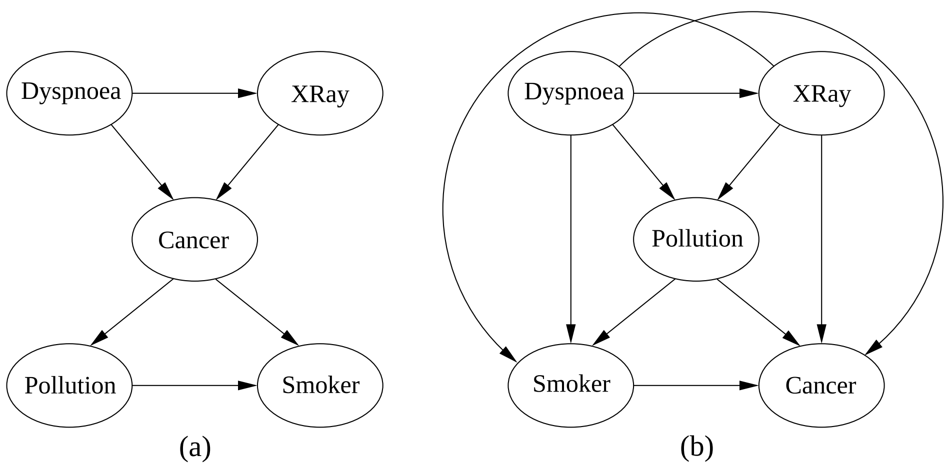
1. Choose the set of relevant variables $\{X_i\}$ that describe the domain.
2. Choose an ordering for the variables, $< X_1, \ldots, X_n >$.
3. While there are variables left:
   (a) Add the next variable X_i to the network.
   (b) Add arcs to the X_i node from some minimal set of nodes already in the net, $Parents(X_i)$, such that the following conditional independence property is satisfied:
   $$P(X_i|X'_1, \ldots, X'_m) = P(X_i|Parents(X_i))$$
   where $X'_1, \ldots, X'_m$ are all the variables preceding X_i, including $Parents(X_i)$.
   (c) Define the CPT for X_i.

Compactness and Node Ordering

- Compactness of BN depends upon sparseness of the system.
- The best order to add nodes is to add the "root causes" first, then the variable they influence, so on until "leaves" reached.

  $\rightarrow$ Causal structure

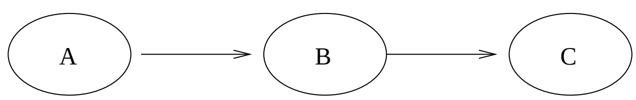
- Alternative structures using different orderings (a) $< D, X, C, P, S >$ (b) $< D, X, P, S, C >$.



1. These BNs still represent same joint distribution.
2. Structure (b) requires as many probabilities as the full joint distribution! See below for *why*.

Conditional Independence: Causal Chains

Causal chains give rise to conditional independence:



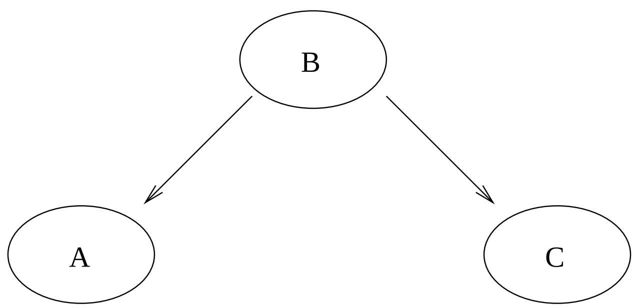
$$P(C|A \wedge B) = P(C|B)$$

Example: "smoking causes cancer which causes dyspnoea"

Weather forecasting example?

Conditional Independence: Common Causes

Common causes (or ancestors) also give rise to conditional independence:



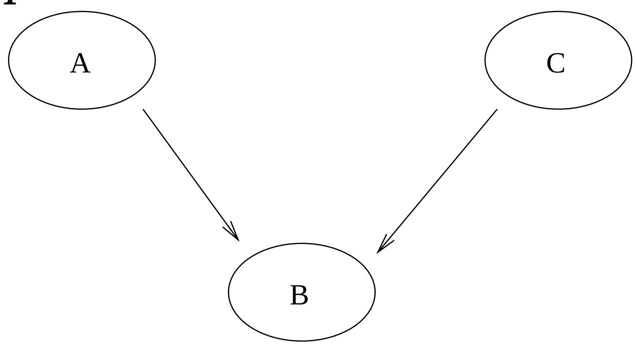
$$P(C|A \wedge B) = P(C|B) \equiv A \perp\!\!\!\perp C|B$$

Example: cancer is a common cause of the two symptoms, a positive XRay result and dyspnoea.

Weather forecasting example?

Conditional Dependence: Common Effects

Common effects (or their descendants) give rise to conditional *dependence:*



$$P(A|C \wedge B) \neq P(A)P(C) \equiv \neg(A \perp\!\!\!\perp C|B)$$

Example: Cancer is a common effect of pollution and smoking.

> Given lung cancer, smoking "explains away" pollution.

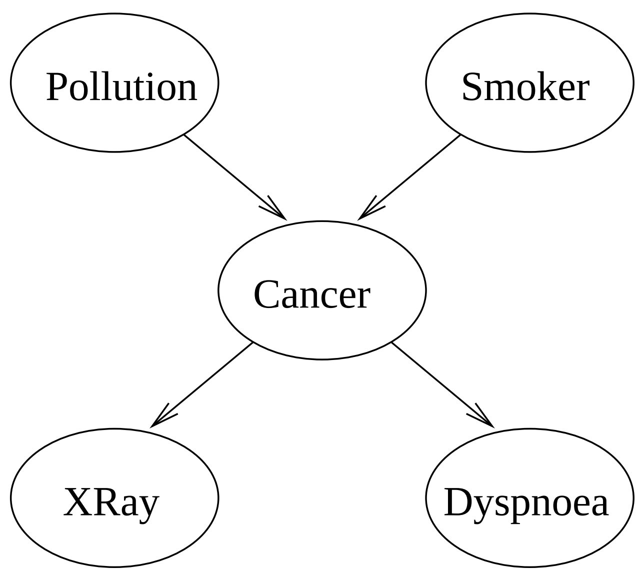
Weather forecasting example?

D-separation

- Graphical criterion of conditional independence.
  $X \perp Y | Z$

- We can determine whether a set of nodes X is independent of another set Y, given a set of evidence nodes E, via the Markov peroperty:
  $X \perp Y | E \rightarrow \quad X \perp\!\!\!\perp Y | E.$

- Example



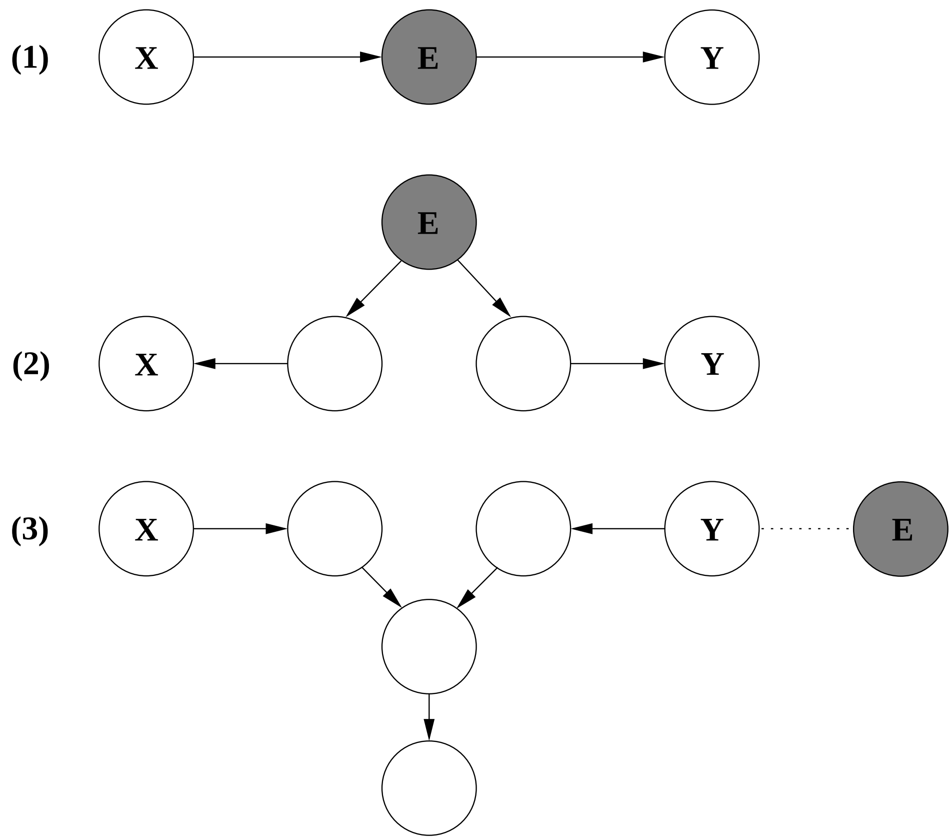
D-separation

How to determine d-separation, $X \perp Y | E$:

- If every undirected path from a node in X to a node in Y is *d-separated* by E, then X and Y are *conditionally independent* given E.
- A set of nodes E *d-separates* two sets of nodes X and Y if every undirected path from a node in X to a node in Y is *blocked* given E.
- A path is *blocked* given a set of nodes E if there is a node Z on the path for which one of three conditions holds:
  1. Z is in E and Z has one arrow on the path leading in and one arrow out (chain).
  2. Z is in E and Z has both path arrows leading out (common cause).
  3. Neither Z nor any descendant of Z is in E, and both path arrows lead in to Z (common effect).

D-separation (cont'd)

- Evidence nodes **E** shown shaded.



Matilda: a visual tool for exploring dependencies

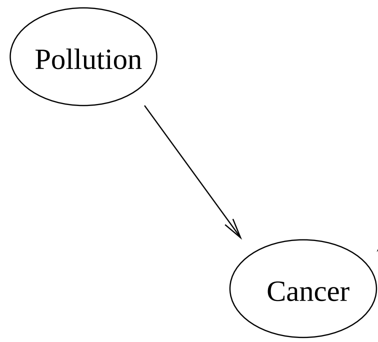
Causal Ordering

Why does variable order affect network density?

Because

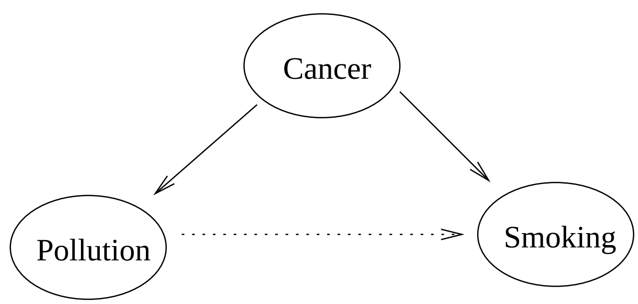
- Using the causal order allows direct representation of conditional independencies
- Violating causal order requires new arcs to re-establish conditional independencies

Causal Ordering (cont'd)



Pollution and *Smoking* are marginally independent.

Given the ordering: Cancer, Pollution, Smoking:



Marginal independence of Pollution and Smoking must be re-established by adding $Pollution \rightarrow Smoking$ or $Smoking \leftarrow Pollution$

Bayesian Networks: Summary

- Bayes' rule allows unknown probabilities to be computed from known ones.
- Conditional independence (due to causal relationships) allows efficient updating
- BNs are a natural way to represent conditional independence info.
  - links between nodes: qualitative aspects;
  - conditional probability tables: quantitative aspects.
- Probabilistic inference: compute the probability distribution for query variables, given evidence variables
- BN Inference is very flexible: can enter evidence about any node and update beliefs in any other nodes.

Inference Algorithms: Overview

- Exact inference
  - Trees and polytrees:
    - message-passing algorithm
  - Multiply-connected networks:
- Approximate Inference
  - Large, complex networks:
    - Stochastic Simulation
    - Other approximation methods
- In the general case, both exact and approximate inference are computationally complex ("NP-hard").
- Causal inference

Inference in chains

Two node network $X \rightarrow Y$.

- Evidence $X = x$, then $Bel(Y) = P(Y|X = x)$ straight from CPT.
- Evidence $Y = y$

$$
\begin{aligned}
Bel(X = x) &= P(X = x|Y = y) \\
&= \frac{P(Y = y|X = x)P(X = x)}{P(y)} \\
&= \alpha P(x)\lambda(x)
\end{aligned}
$$

where

$$\alpha = \frac{1}{P(Y = y)}$$

$P(x)$ is the prior, and $\lambda(x) = P(Y = y|X = x)$ is the **likelihood**.

Since $\sum_i Bel(Y = y_i) = 1$, we can compute α as a **normalizing constant**.

Example: $Flu \rightarrow\ HighTemp$

Suppose $P(Flu = T) = 0.05$,
$P(HighTemp = T|Flu = T) = 0.9$,
$P(HighTemp = T|Flu = F) = 0.2$.

Given evidence $HighTemp = T$, then

$$\begin{aligned}
Bel(Flu = T) &= \alpha P(Flu = T)\lambda(Flu = T) \\
&= \alpha \times 0.05 \times 0.9 = \alpha 0.045 \\
Bel(Flu = F) &= \alpha P(Flu = F)\lambda(Flu = F) \\
&= \alpha \times 0.95 \times 0.2 = \alpha 0.19
\end{aligned}$$

We can compute α via

$$Bel(Flu = T) + Bel(Flu = F) = 1 = \alpha 0.045 + \alpha 0.19$$

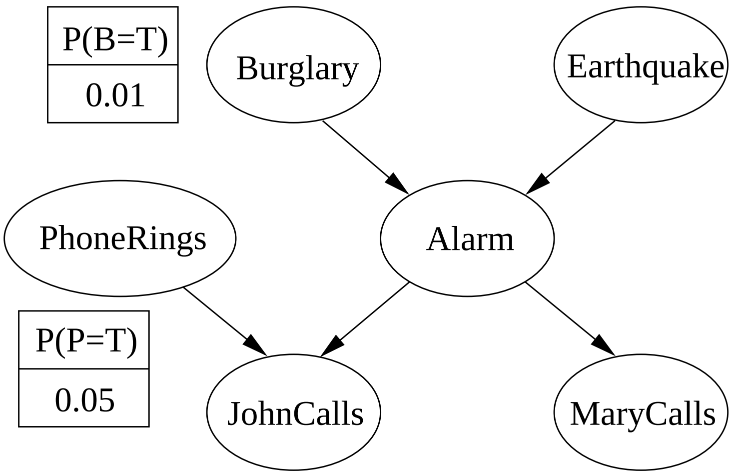
giving

$$\alpha = \frac{1}{0.19 + 0.045}$$

This allows us to finish the belief update:

$$\begin{aligned}
Bel(Flu = T) &= \frac{0.045}{0.19 + 0.045} = 0.8085 \\
Bel(Flu = F) &= \frac{0.19}{0.19 + 0.045} = 0.1915
\end{aligned}$$

Earthquake example



P(B=T)
0.01

P(E=T)
0.02

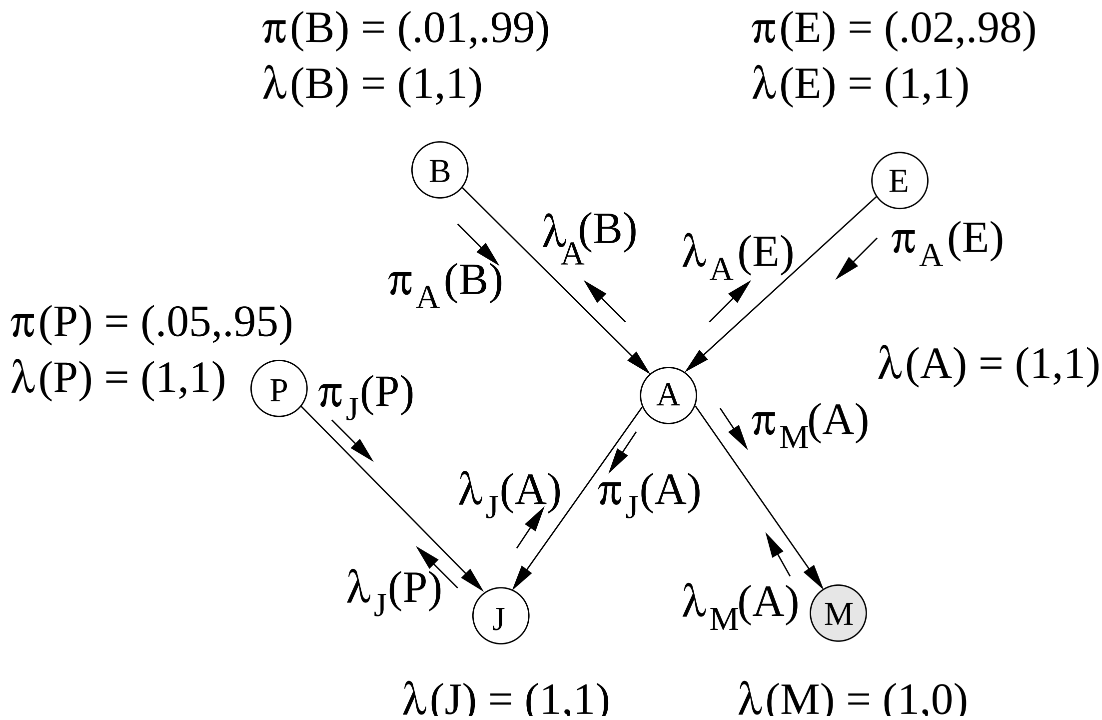
P(P=T)
0.05

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

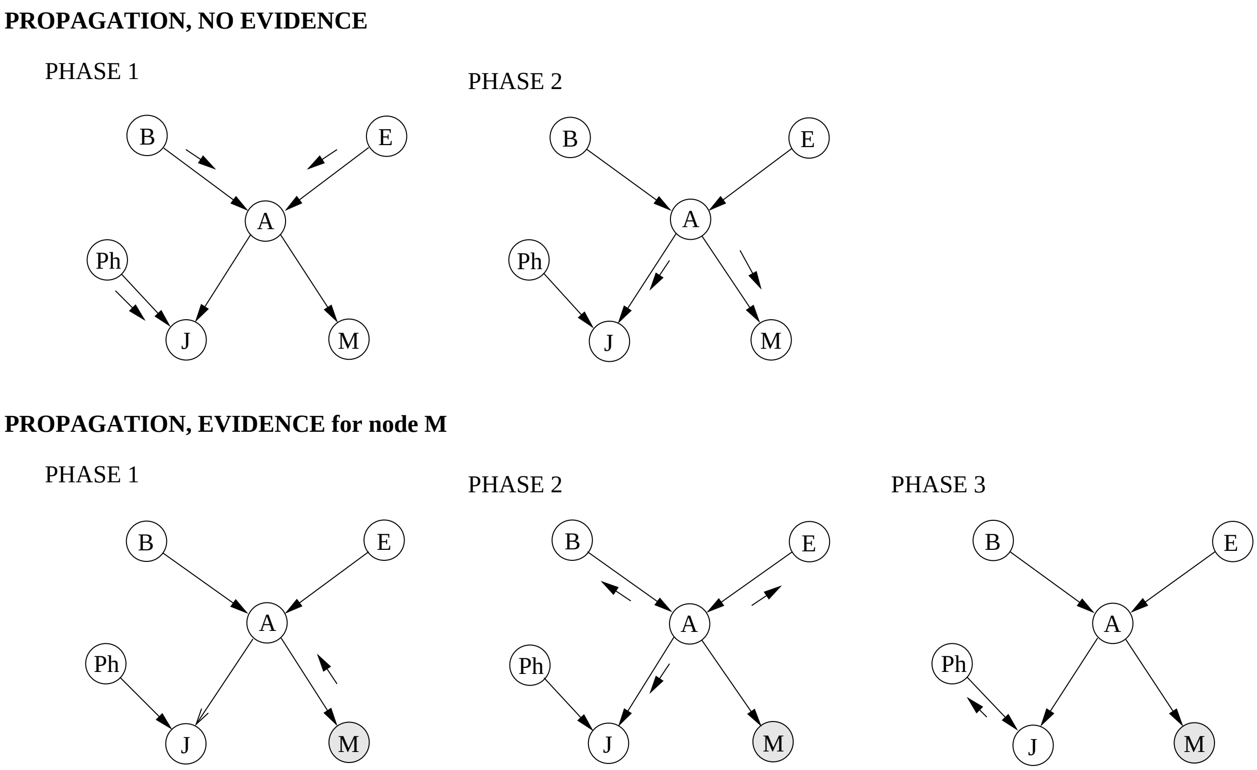
P	A	P(J=T|P,A)
T	T	0.95
T	F	0.50
F	T	0.90
F	F	0.01

A	P(M=T|A)
T	0.70
F	0.01

Inference in polytrees: message passing



Message propagation



Message-passing algorithm: features

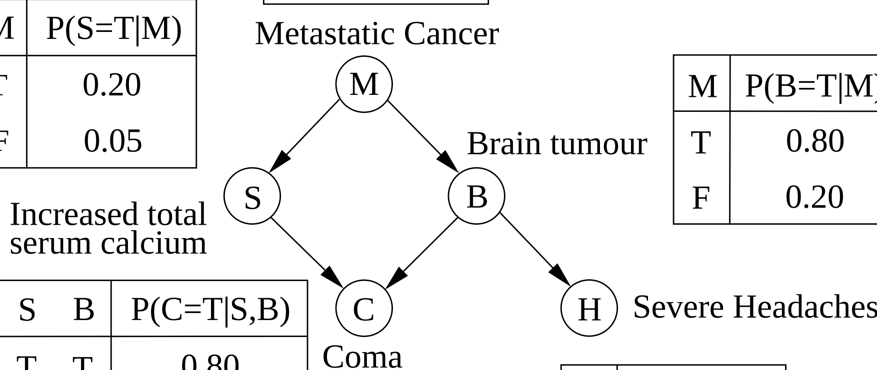
- All computations are local $\Rightarrow$ efficient
- Requires summation over all joint instantiations of parent nodes $\Rightarrow$ exponential in no. of parents.
- No. of propagation steps depends on length of longest path

Inference in multiply connected networks

- Networks where two nodes are connected by more than one path
  - Two or more possible causes which share a common ancestor
  - One variable can influence another through more than one causal mechanism
- Example: Cancer network

P(M=T) = 0.9

M	P(S=T|M)
T	0.20
F	0.05



M	P(B=T|M)
T	0.80
F	0.20

S	B	P(C=T|S,B)
T	T	0.80
T	F	0.80
F	T	0.80
F	F	0.05

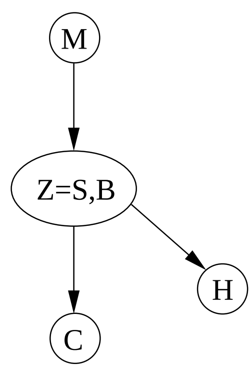
B	P(H=T|B)
T	0.80
F	0.60

- Message passing doesn't work - evidence gets "counted twice"

Clustering methods

- Transform network into a probabilistically equivalent polytree by merging (clustering) offending nodes
- Cancer example: new node Z combining B and C

P(M=T)
0.9



M	Z	P(Z|M)
T	TT	0.16
	TF	0.64
	FT	0.16
	FF	0.16

Z	P(H=T|Z)
TT	0.8
TF	0.6
FT	0.8
FF	0.6

Z	P(C=T|Z)
TT	0.8
TF	0.6
FT	0.8
FF	0.05

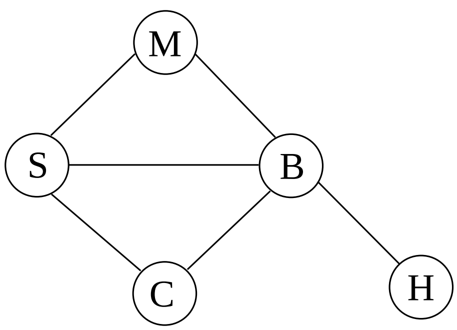
$$P(z|a) = P(b,c|a) = P(b|a)P(c|a)$$

$$P(e|z) = P(e|b,c) = P(e|c)$$

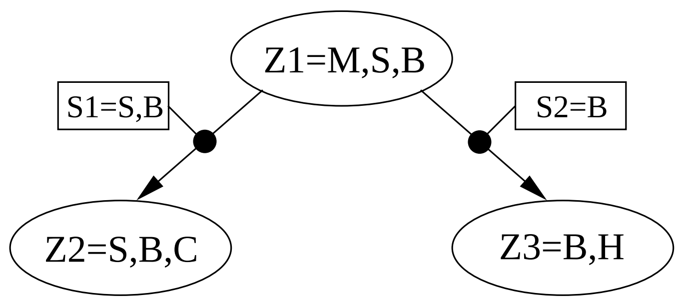
$$P(d|z) = P(d|b,c)$$

Jensen join-tree method

- Jensen Join-tree (Jensen, 1996) version the current most efficient algorithm in this class (e.g. used in Hugin, Netica).



(a)



(b)

Jensen join-tree method (cont.)

Network evaluation done in two stages

- Compile into join-tree
  - May be slow
  - May require too much memory if original network is highly connected
- Do belief updating in join-tree (usually fast)

Caveat: clustered nodes have increased complexity; updates may be computationally complex

Approximate inference with stochastic simulation

- Use the network to generate a large number of cases that are consistent with the network distribution.
- Evaluation may not converge to exact values (in reasonable time).
- Usually converges to close to exact solution quickly if the evidence is not too unlikely.
- Performs better when evidence is nearer to root nodes, however in real domains, evidence tends to be near leaves (Nicholson&Jitnah, 1998)
- Not available in Netica, is available in Hugin and other BN software.

Causal modeling

We should like to do causal modeling with our Bayesian networks.

Prerequisite: arcs are truly causal (hence, nodes are properly ordered).

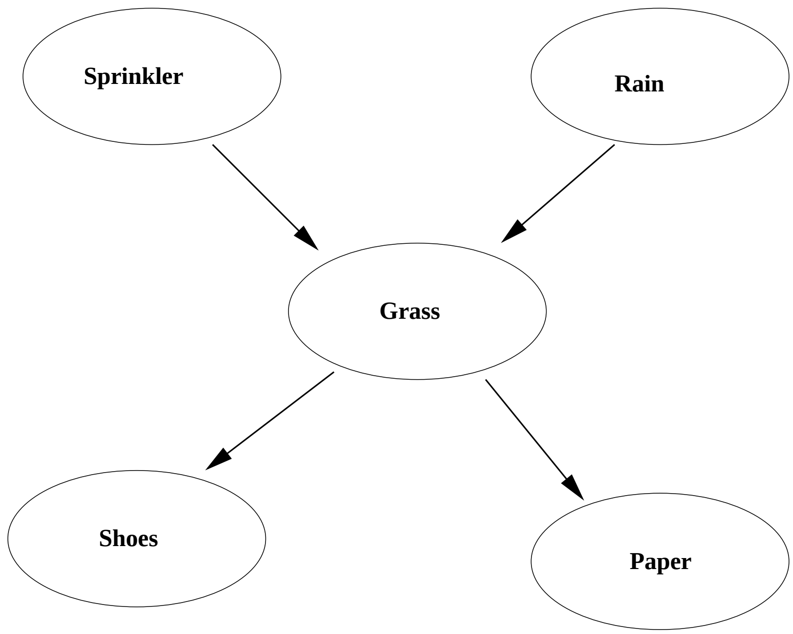
Reasoning about real or hypothetical interventions:

- what if we upgrade quality in manufacturing?
- what if we treat the patient with X, Y, Z?

For planning, control, prediction.

Common practice appears to be: let observation stand for intervention.

Causal inference



If we observe that the lawn is wet:

- We can infer in any direction; everything updates
- We get, e.g., “explaining away” between causes

What happens if we *intervene* in a causal process?

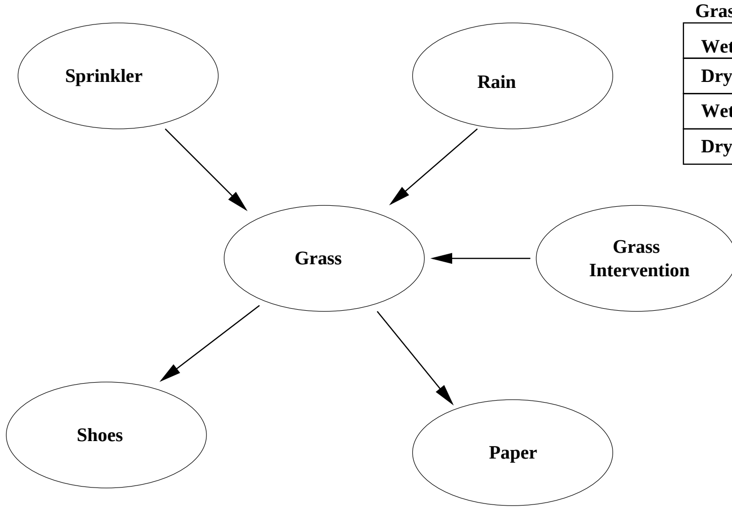
Spirtes, et al., (1993), Pearl (2000) answer: cut links to parents and *then* update.

- No explaining away; parents are then unaffected
- Downstream updating is as normal

Causal inference

We prefer (conceptually) to augment the graph with an intervention variable:

	Interevene		
Grass	T	F	Rain ...
Wet	99.9	x	
Dry	0.1	y	
Wet	99.9	x	
Dry	0.1	y	



- Simplistically, parent connections are severed
- With full generality, X acquires a new parent D_X
  - Allows any degree of control for intervention
  - Allows any kind of interaction with existing parents
  - Bayesian update algorithms unaffected

Inference: Summary

- Probabilistic inference: compute the probability distribution for query variables, given evidence variables
- Causal inference: compute the probability distribution for query variables, given intervention
- BN Inference is very flexible: can enter evidence about any node and update beliefs in any other nodes.
- The speed of inference in practice depends on the structure of the network: how many loops; numbers of parents; location of evidence and query nodes.
- BNs can be used to model causal intervention.

Extensions to Bayesian Networks

- For decision making: Bayesian decision networks
- For reasoning about changes over time: dynamic Bayesian networks

Making Decisions

- Bayesian networks can be extended to support decision making.
- **Preferences** between different outcomes of various plans.
  - Utility theory
- **Decision theory** = Utility theory + Probability theory.

Expected Utility

$$EU(A|E) = \sum_i P(O_i|E, A)\, U(O_i|A) \tag{1}$$

- E = available evidence,
- A = a nondeterministic action
- O_i = possible outcome state
- U = utility

Decision Networks

A Decision network represents information about

- the agent's current state
- its possible actions
- the state that will result from the agent's action
- the utility of that state

Also called, *Influence Diagrams* (Howard&Matheson, 1981).

Type of Nodes

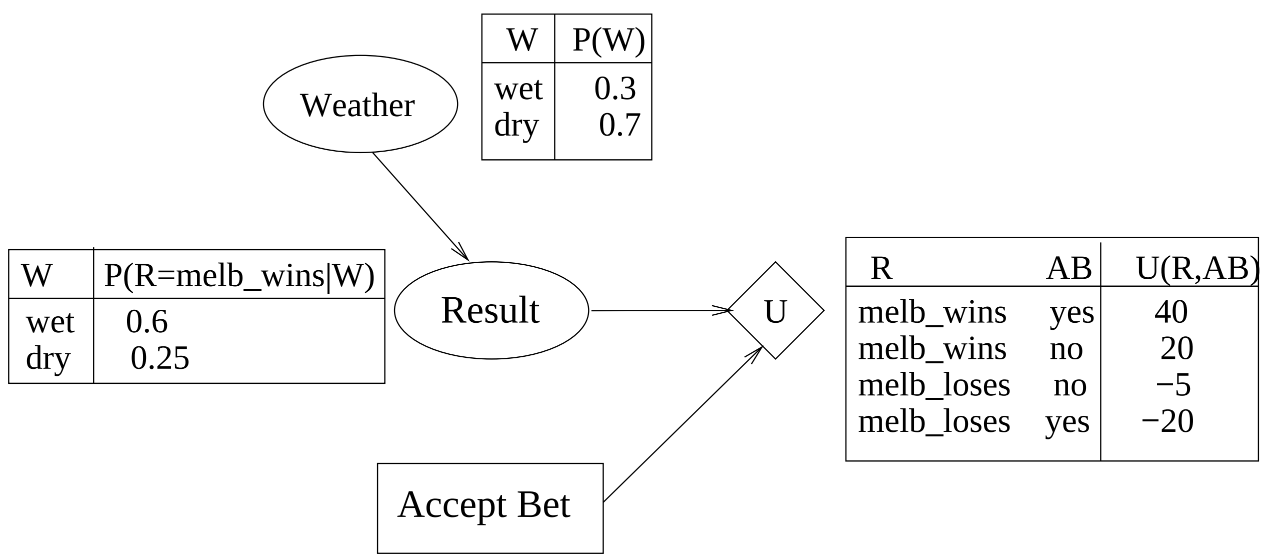
Chance nodes: (ovals) represent random variables (same as Bayesian networks). Has an associated CPT. Parents can be decision nodes and other chance nodes.

Decision nodes: (rectangles) represent points where the decision maker has a choice of actions.

Utility nodes: (diamonds) represent the agent's utility function (also called **value nodes** in the literature). Parents are variables describing the outcome state that directly affect utility. Has an associated table representing multi-attribute utility function.

Example: Football Team

Clare's football team, Melbourne, is going to play her friend John's team, Carlton. John offers Clare a friendly bet: whoever's team loses will buy the wine next time they go out for dinner. They never spend more than \$15 on wine when they eat out. When deciding whether to accept this bet, Clare will have to assess her team's chances of winning (which will vary according to the weather on the day). She also knows that she will be happy if her team wins and miserable if her team loses, regardless of the bet.



W	P(W)
wet	0.3
dry	0.7

W	P(R=melb_wins|W)
wet	0.6
dry	0.25

R	AB	U(R,AB)
melb_wins	yes	40
melb_wins	no	20
melb_loses	no	−5
melb_loses	yes	−20

Evaluating Decision Networks: Algorithm

1. Add any available evidence.
2. For each action value in the decision node:
   (a) Set the decision node to that value;
   (b) Calculate the posterior probabilities for the parent nodes of the utility node, as for Bayesian networks, using a standard inference algorithm;
   (c) Calculate the resulting expected utility for the action.
3. Return the action with the highest expected utility.

Evaluating Decision Networks: Example

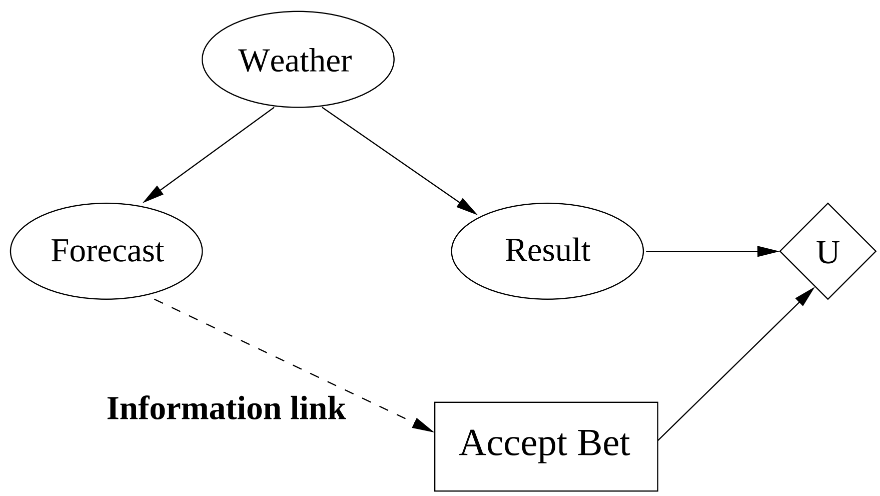
$$
\begin{aligned}
P(R = melb\_wins) &= P(W = w)P(R = melb\_wins|W = w) \\
&= +P(W = d)P(R = melb\_wins|W = d)
\end{aligned}
$$

$$
\begin{aligned}
EU(AB = yes) &= P(R = wins)U(R = wins|AB = yes) \\
&+ P(R = loses)U(R = loses|AB = yes) \\
&= (0.3 \times 0.6 + 0.7 \times 0.25)40 \\
&+ (0.3 \times 0.4 + 0.7 \times 0.75) - 20 \\
&= 0.355 \times 40 + 0.645 \times -20 = 14.2 - 12.9 \\
&= 1.3 \\
EU(AB = no) &= P(R = wins)U(R = wins|AB = no) \\
&+ P(R = loses)U(R = loses|AB = no) \\
&= (0.3 \times 0.6 + 0.7 \times 0.25)20 \\
&+ (0.3 \times 0.4 + 0.7 \times 0.75) - 5 \\
&= 0.355 \times 20 + 0.645 \times -5 = 7.1 - 3.225 \\
&= 3.875
\end{aligned}
$$

Information Links

- Indicate when a chance node needs to be observed before a decision is made.

W	F	P(F|W)
wet	rainy	0.60
	cloudy	0.25
	sunny	0.15
dry	rainy	0.10
	cloudy	0.40
	sunny	0.50



Decision Table

F	Accept Bet
rainy	yes
cloudy	no
sunny	no

Decision Table Algorithm

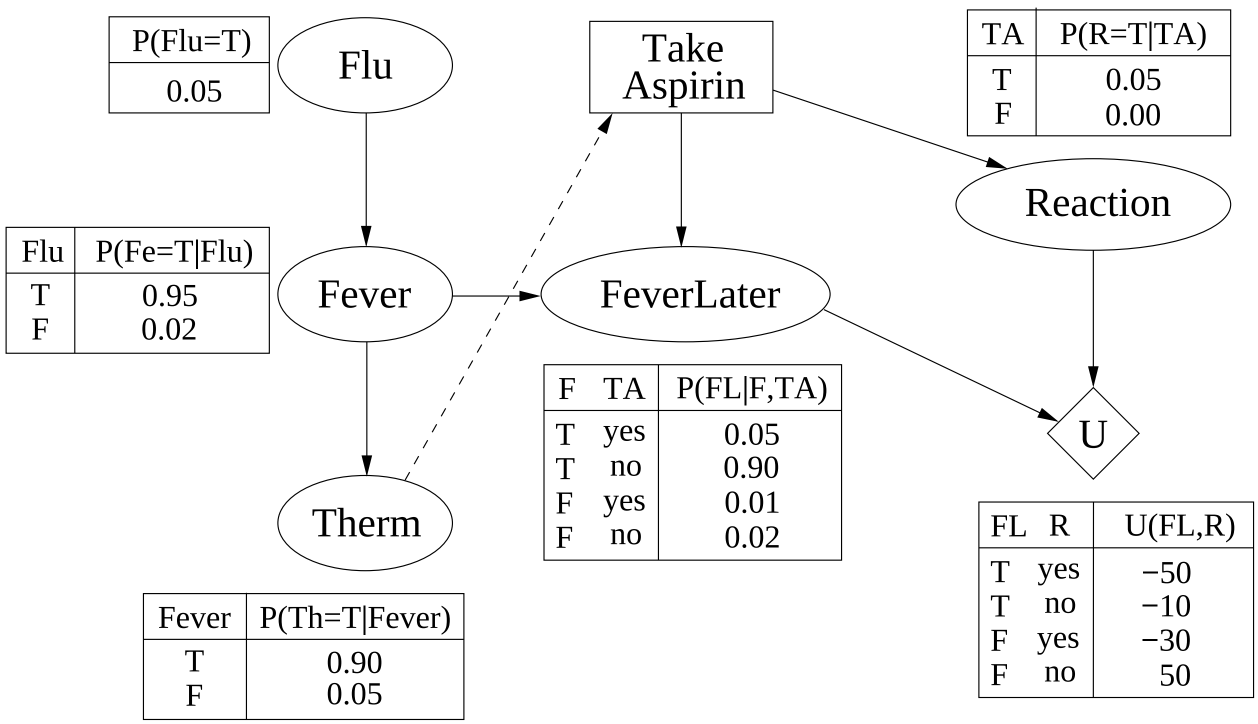
1. Add any available evidence.
2. For each combination of values of the parents of the decision node:
   (a) For each action value in the decision node:
      i. Set the decision node to that value;
      ii. Calculate the posterior probabilities for the parent nodes of the utility node, as for Bayesian networks, using a standard inference algorithm;
      iii. Calculate the resulting expected utility for the action.

   (b) Record the action with the highest expected utility in the decision table.
3. Return the decision table.

Fever problem description

Suppose that you know that a fever can be caused by the flu. You can use a thermometer, which is fairly reliable, to test whether or not you have a fever. Suppose you also know that if you take aspirin it will almost certainly lower a fever to normal. Some people (about 5% of the population) have a negative reaction to aspirin. You'll be happy to get rid of your fever, as long as you don't suffer an adverse reaction if you take aspirin.

Fever decision network



P(Flu=T)
0.05

Flu	P(Fe=T|Flu)
T	0.95
F	0.02

Fever	P(Th=T|Fever)
T	0.90
F	0.05

TA	P(R=T|TA)
T	0.05
F	0.00

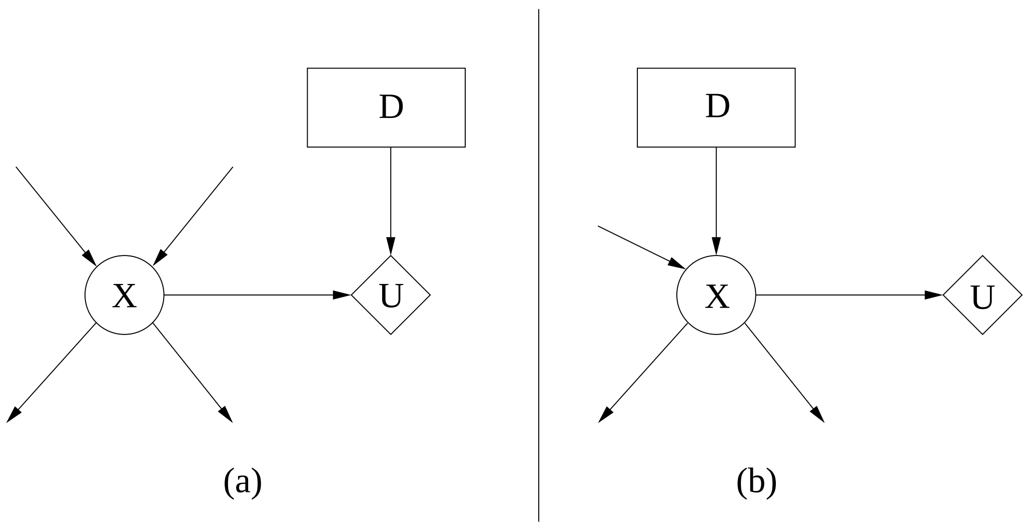
F	TA	P(FL|F,TA)
T	yes	0.05
T	no	0.90
F	yes	0.01
F	no	0.02

FL	R	U(FL,R)
T	yes	−50
T	no	−10
F	yes	−30
F	no	50

Fever decision table

Ev.	*Bel(FL=T)*	EU(TA=yes)	EU(TA=no)	Dec.
None	0.046	45.27	**45.29**	no
Th=F	0.525	45.41	**48.41**	no
Th=T	0.273	**44.1**	19.13	yes
Th=T & *Re=T*	0.273	-30.32	**0**	no

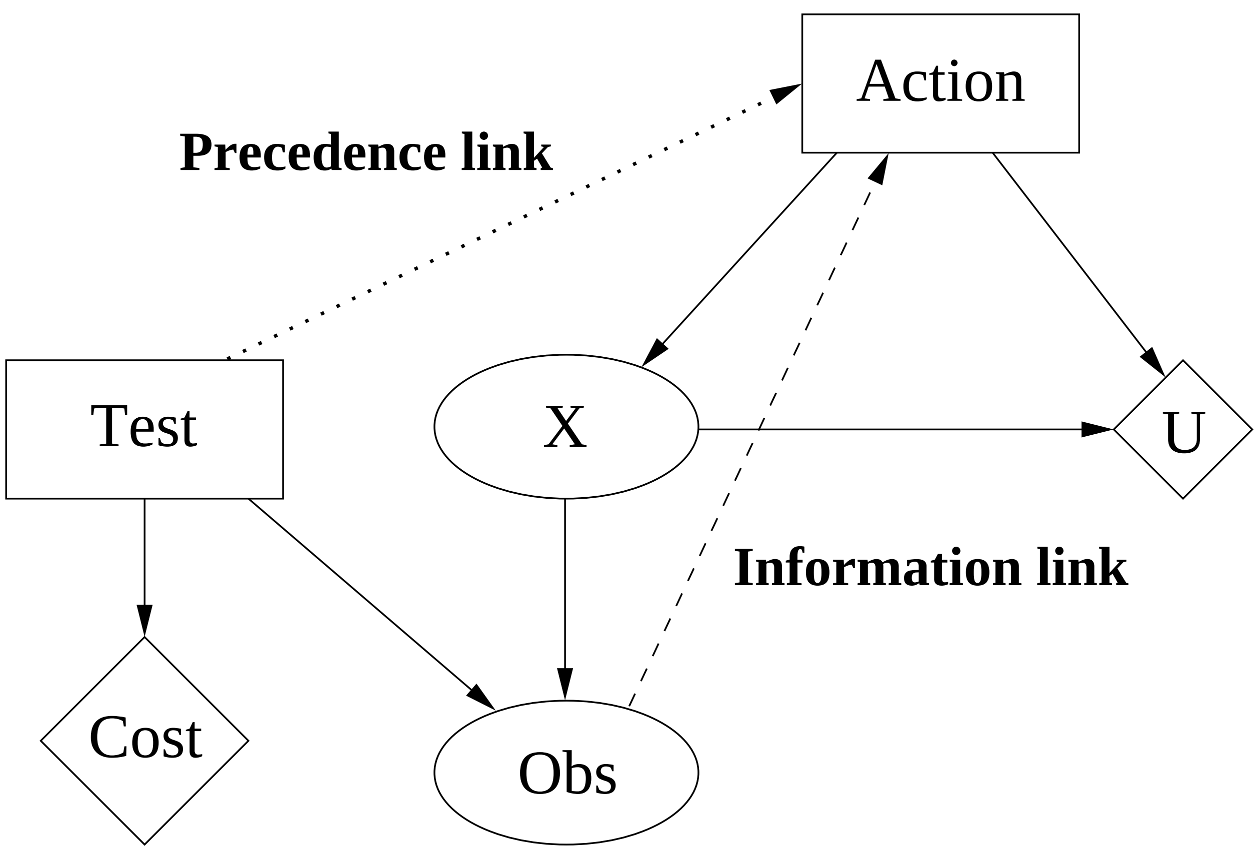
Types of actions



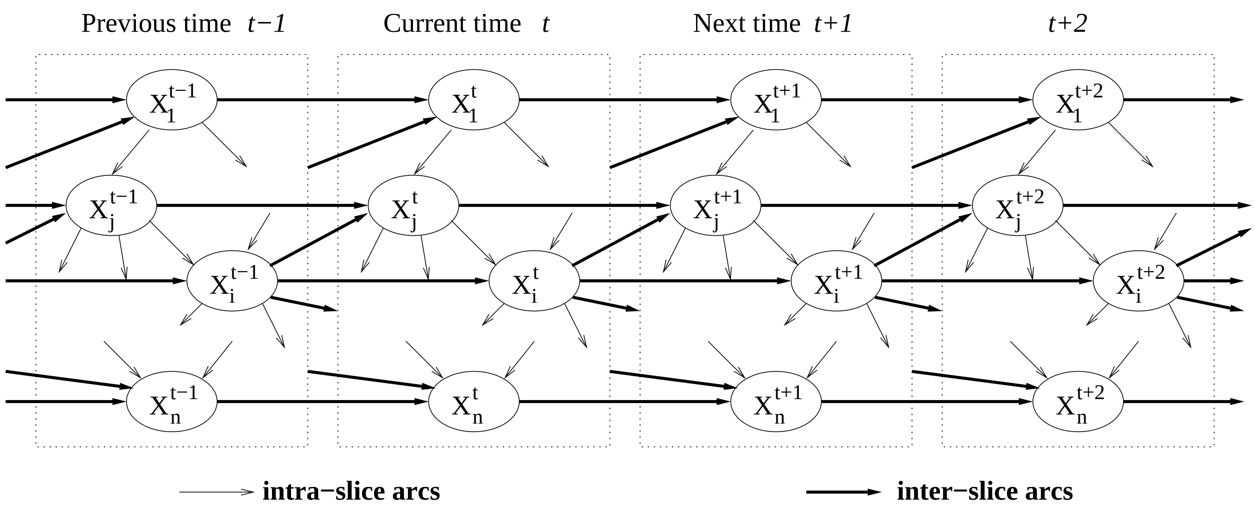
(a) Non-intervening and (b) Intervening

Sequential decision making

- Precedence links used to show temporal ordering.
- Network for a test-action decision sequence

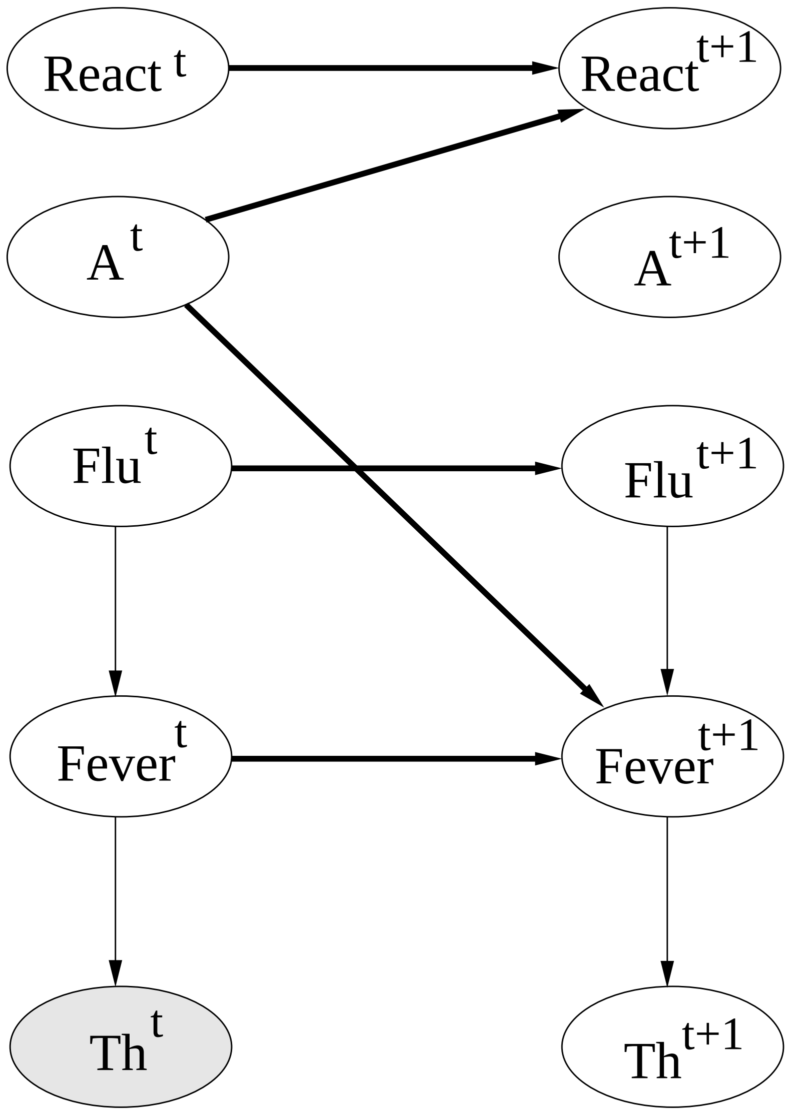


Dynamic Belief Networks



- One node for each variable for each time step.
- **Intra-slice** arcs $X_i^T \longrightarrow X_j^T$
- **Inter-slice (temporal)** arcs
  1. $X_i^T \longrightarrow X_i^{T+1}$
  2. $X_i^T \longrightarrow X_j^{T+1}$

Fever DBN

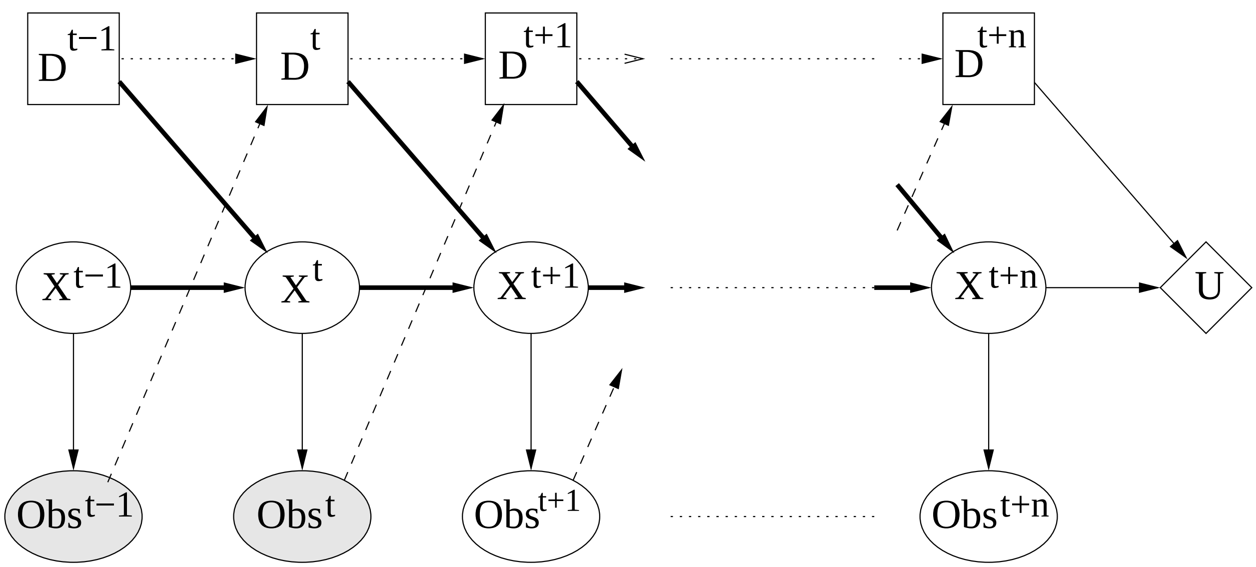


DBN reasoning

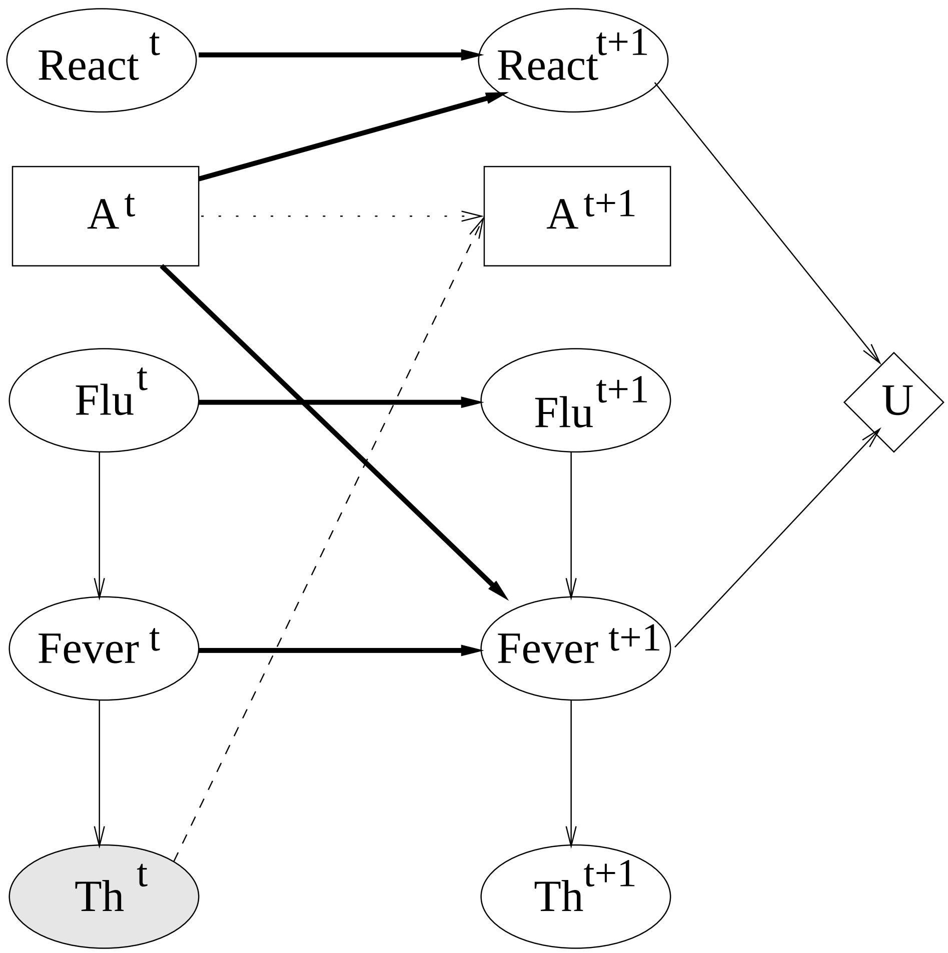
- Can calculate distributions for S_{t+1} and further: **probabilistic projection**.
- Reasoning can be done using standard BN updating algorithms
- This type of DBN gets very large, very quickly.
- Usually only keep two time slices of the network.

Dynamic Decision Network

- Similarly, Decision Networks can be extended to include temporal aspects.
- Sequence of decisions taken = Plan.



Fever DDN



Extensions: Summary

- BNs can be extended with decision nodes and utility nodes to support decision making: *Decision Networks* or *Influence Diagrams*.
- BNs and decision networks can be extended to allow explicit reasoning about changes over time.

Uses of Bayesian Networks

1. Calculating the belief in query variables given values for evidence variables.

2. Predicting values in dependent variables given values for independent variables.

3. Modeling causal interventions.

4. Decision making based on probabilities in the network and on the agent's utilities (Influence Diagrams [Howard and Matheson 1981]).

5. Deciding which additional evidence should be observed in order to gain useful information (see KEBN section below).

6. *Sensitivity analysis* to test impact of changes in probabilities or utilities on decisions (see KEBN section below).

NEXT...

(AFTER LUNCH)

Lab session using Netica

Room 1XX, First Floor

LUNCH!

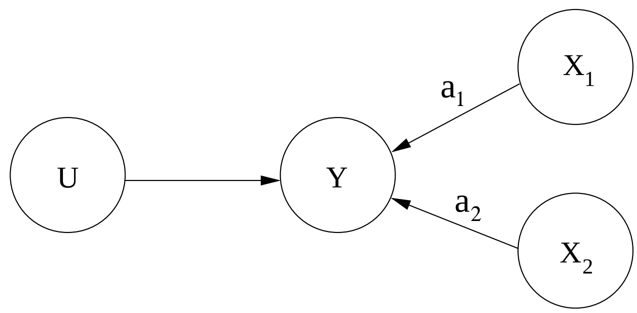
Learning Bayesian Networks

- Linear and Discrete Models
- Learning Network Parameters
  - Linear Coefficients
  - Learning Probability Tables
- Learning Causal Structure
- Conditional Independence Learning
  - Statistical Equivalence
  - TETRAD II
- Bayesian Learning of Bayesian Networks
  - Cooper & Herskovits: K2
  - Learning Variable Order
  - Statistical Equivalence Learners
- Full Causal Learners
- Minimum Encoding Methods
  - Lam & Bacchus's MDL learner
  - MML metrics
  - MML search algorithms
  - MML Sampling
- Empirical Results

Linear and Discrete Models

Linear Models: Used in biology & social sciences since Sewall Wright (1921)

Linear models represent causal relationships as sets of linear functions of "independent" variables.



Equivalently:

$$X_3 = a_{13} X_1 + a_{23} X_2 + \epsilon_1$$

Structural equation models (SEMs) are close relatives

Discrete models: "Bayesian nets" replace vectors of linear coefficients with CPTs.

Learning Linear Parameters

Maximum likelihood methods have been available since Wright's path model analysis (1921).

Equivalent methods:

- Simon-Blalock method (Simon, 1954; Blalock, 1964)
- Ordinary least squares multiple regression (OLS)

Learning Conditional Probability Tables

Spiegelhalter & Lauritzen (1990):

- assume parameter independence
- each CPT cell i = a parameter in a Dirichlet distribution

$$D[\alpha_1, \ldots, \alpha_i, \ldots, \alpha_K]$$

  for K parents

- prob of outcome i is $\alpha_i / \Sigma_{k=1}^{K} \alpha_k$
- observing outcome i update D to

$$D[\alpha_1, \ldots, \alpha_i + 1, \ldots, \alpha_K]$$

Others are looking at learning without parameter independence. E.g.,

- Decision trees to learn structure within CPTs (Boutillier et al. 1996).
- Dual log-linear and full CPT models (Neil, Wallace, Korb 1999).

Learning Causal Structure

This is the *real* problem; parameterizing models is relatively straightforward estimation problem.

Size of the dag space is superexponential:

- Number of possible orderings: $n!$
- Times number of ways of pairing up (for arcs): $2^{C_2^n}$
- Minus number of possible cyclic graphs

Without the subtraction (which is a small proportion):

n	$n!2^{C_2^n}$
0	0
1	1
2	4
3	48
4	1536
5	12280
10	127677049435953561600
100	[too many digits to show]

Learning Causal Structure

There are two basic methods:

- Learning from conditional independencies (CI learning)
- Learning using a scoring metric (Metric learning)

CI learning (Verma and Pearl, 1991)

Suppose you have an Oracle who can answer yes or no to any question of the type:

$$X \perp\!\!\!\perp Y | \mathbf{S}?$$

(i.e., is X conditional independence Y given $\mathbf{S}$)

Then you can learn the correct causal model, up to statistical equivalence (patterns).

Verma-Pearl Algorithm

two rules allow discovery of the set of causal models consistent with all such answers ("patterns"):

1. **Principle I** Put an undirected link between any two variables X and Y iff
   for every $\mathbf{S}$ s.t. $X, Y \notin \mathbf{S}$

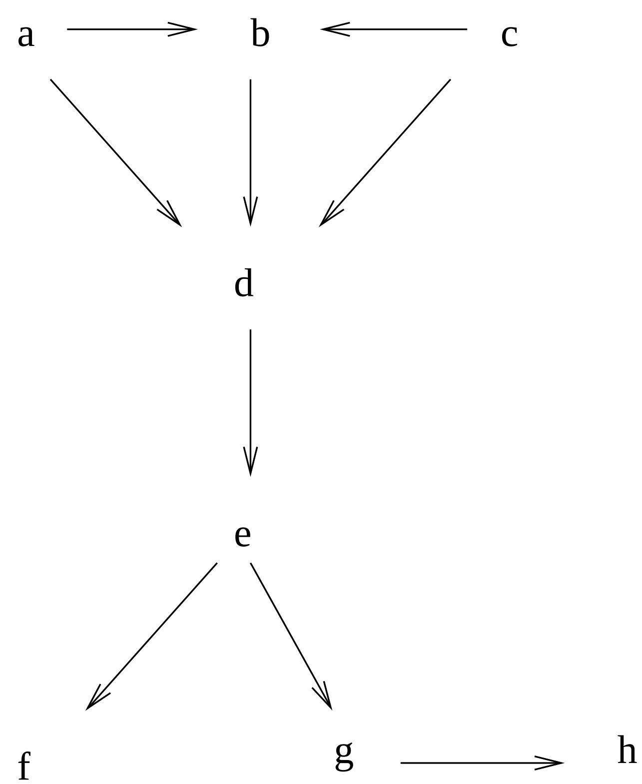
   $$\neg(X \perp\!\!\!\perp Y)|\mathbf{S}$$

2. **Principle II** For every undirected v-structure $X - Z - Y$ orient the arcs $X \rightarrow Z \leftarrow Y$ iff

   $$\neg(X \perp\!\!\!\perp Y)|\mathbf{S}$$

   for **every** $\mathbf{S}$ s.t. $X, Y \notin \mathbf{S}$ and $Z \in \mathbf{S}$.

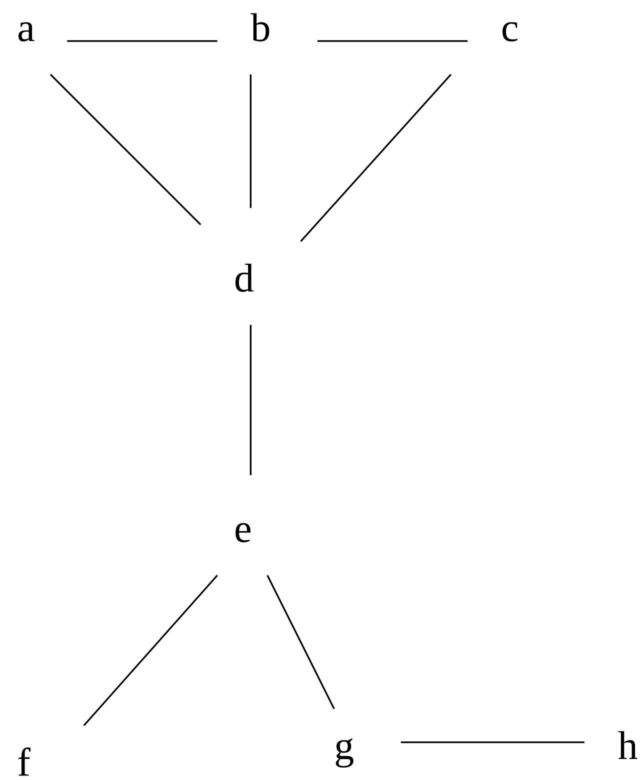
CI learning example



CI learning example

1) $a - b - c$	$a \rightarrow b \leftarrow c$
	b[induces a dependency]
2) $a - d - c$	$a \rightarrow d \leftarrow c$
3) $c - d - e$	$\neg(c \rightarrow d \leftarrow e)$
	therefore $c \rightarrow d \rightarrow e$
4) $a - d - e$	no news
5) $b - d - e$	no news
6) $d - e - f$	$\neg(d \rightarrow e \leftarrow f)$
7) $d - e - g$	$\neg(d \rightarrow e \leftarrow g)$
6) $e - g - h$	$\neg(e \rightarrow g \leftarrow h)$

CI learning example



Statistical Equivalence

Verma and Pearl's rules identify the set of causal models which are statistically equivalent —

> Two causal models H_1 and H_2 are **statistically equivalent** iff they contain the same variables and joint samples over them provide no statistical grounds for preferring one over the other.

Examples

- All fully connected models are equivalent.
- A $\rightarrow$ B $\rightarrow$ C and A $\leftarrow$ B $\leftarrow$ C.
- A $\rightarrow$ B $\rightarrow$ D $\leftarrow$ C and A $\leftarrow$ B $\rightarrow$ D $\leftarrow$ C.

Statistical Equivalence

- (Verma and Pearl, 1991): Any two causal models over the same variables which have the same skeleton (undirected arcs) and the same directed v-structures are statistically equivalent.

- Chickering (1995): If H_1 and H_2 are statistically equivalent, then they have the same maximum likelihoods relative to any joint samples:

  $$\max P(e|H_1, \theta_1) = \max P(e|H_2, \theta_2)$$

  where θ_i is a parameterization of H_i

TETRAD II

— Spirtes, Glymour and Scheines (1993)

Replace the Oracle with statistical tests:

- for linear models a significance test on partial correlation

$$X \perp\!\!\!\perp Y | \mathbf{S} \text{ iff } \rho_{XY \cdot \mathbf{S}} = 0$$

- for discrete models a χ^2 test on the difference between CPT counts expected with independence (E_i) and observed (O_i)

$$X \perp\!\!\!\perp Y | \mathbf{S} \text{ iff } \sum_i O_i \ln \left(\frac{O_i}{E_i} \right)^2 \approx 0$$

Implemented in their **PC Algorithm**

TETRAD II: Weak Links and Small Samples

Main weakness of TETRAD II: orthodox sig tests.

- As the order of partials goes up, the number of correlations required to be estimated goes up.
- Since sig tests are not robust, TETRAD II may work ok on small models with large samples, but unlikely to work on large models with modest samples
- This point was demonstrated empirically in Dai, Korb, Wallace & Wu (1997).

Bayesian LBN: Cooper & Herskovits' K2

— Cooper & Herskovits (1991, 1992)

Compute $P(h_i|e)$ by brute force, under the assumptions:

1. All variables are discrete.
2. Samples are i.i.d.
3. No missing values.
4. All values of child variables are uniformly distributed.
5. Priors over hypotheses are uniform.

With these assumptions, Cooper & Herskovits reduce the computation of $P_{CH}(h, e)$ to a polynomial time counting problem.

Cooper & Herskovits

But the hypothesis space is exponential; they go for dramatic simplification:

6. Assume we know the temporal ordering of the variables.

In that case, for any pair of variables the only problem is

- deciding whether they are connected by an arc
  $\rightarrow$ arc direction is trival
  $\rightarrow$ cycles are impossible.

New hypothesis space has size only $2^{(n^2-n)/2}$ (still exponential).

Algorithm "K2" does a greedy search through this reduced space.

Learning Variable Order

Reliance upon a given variable order is a major drawback to K2

> And many other algorithms (Buntine 1991, Bouckert 1994, Suzuki 1996, Madigan & Raftery 1994)

What's wrong with that?

- We want autonomous AI (data mining). If experts can order the variables they can likely supply models.
- Determining variable ordering is half the problem. If we know A comes before B, the only remaining issue is whether there is a link between the two.
- The number of orderings consistent with dags is exponential (Brightwell & Winkler 1990; number complete). So iterating over all possible orderings will not scale up.

Statistical Equivalence Learners

Heckerman & Geiger (1995) advocate learning only up to statistical equivalence classes (a la TETRAD II).

> Since observational data cannot distinguish btw equivalent models, there's no point trying to go futher.
>
> $\Rightarrow$ Madigan, Andersson, Perlman & Volinsky (1996) follow this advice, use uniform prior over equivalence classes.
>
> $\Rightarrow$ Geiger and Heckerman (1994) define Bayesian metrics for linear and discrete equivalence classes of models (BGe and BDe)

GES

Greedy Equivalence Search (GES)

- Product of the CMU-Microsoft group (Meek, 1996; Chickering, 2002)
- Two-stage greedy search: Begin with unconnected pattern
  1. Greedily add single arcs until reaching a local maximum
  2. Prune back edges which don't contribute to the score
- Uses a Bayesian score over patterns only
- Implemented in TETRAD and Murphy's BNT

Statistical Equivalence Learners

Wallace & Korb (1999): This is not right!

- These are **causal** models; they *are* distinguishable on *experimental* data.
  - Failure to collect some data is no reason to change prior probabilities.
    E.g., If your thermometer topped out at 35°, you wouldn't treat $\geq 35^\circ$ and 34° as equally likely.
- Not all equivalence classes are created equal:
  $\{$ A $\leftarrow$ B $\rightarrow$ C, A $\rightarrow$ B $\rightarrow$ C, A $\leftarrow$ B $\leftarrow$ C$\}$
  $\{$ A $\rightarrow$ B $\leftarrow$ C$\}$
- *Within* classes some dags should have greater priors than others... E.g.,
  LightsOn $\rightarrow$ InOffice $\rightarrow$ LoggedOn v.
  LightsOn $\leftarrow$ InOffice $\rightarrow$ LoggedOn

Full Causal Learners

So... a full causal learner is an algorithm that:

1. Learns causal connectedness.
2. Learns v-structures.
   Hence, learns equivalence classes.
3. Learns full variable order.
   Hence, learns full causal structure (order + connectedness).

- TETRAD II: 1, 2.
- Madigan et al.; Heckerman & Geiger (BGe, BDe): 1, 2.
- GES: 1, 2.
- Cooper & Herskovits' K2: 1.
- Lam and Bacchus MDL: 1, 2 (partial), 3 (partial).
- Wallace, Neil, Korb MML: 1, 2, 3.

CaMML

Minimum Message Length (Wallace & Boulton 1968) uses Shannon's measure of information:

$$I(m) = -\log P(m)$$

Applied in reverse, we can compute $P(h, e)$ from $I(h, e)$.

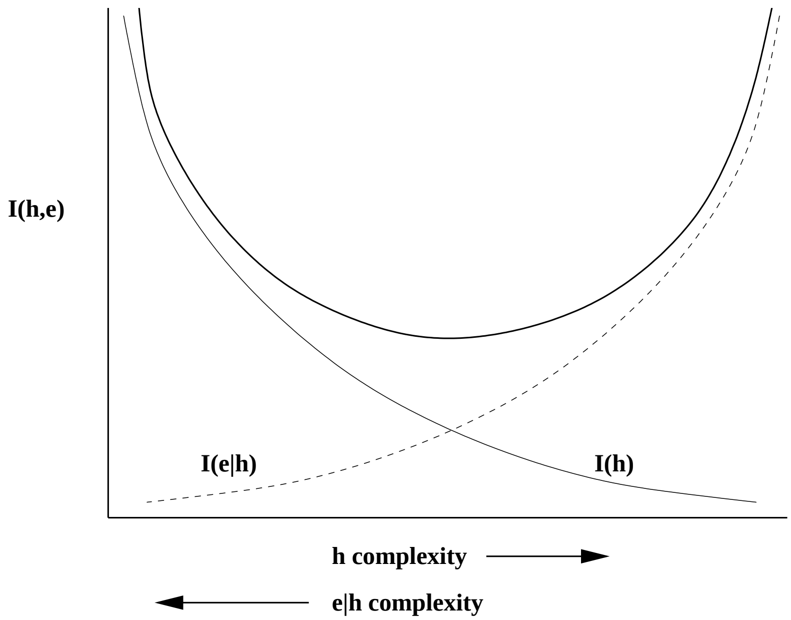
Given an *efficient* joint encoding method for the hypothesis & evidence space (i.e., satisfying Shannon's law), MML:

> Searches $\{h_i\}$ for that hypothesis h that minimizes $I(h) + I(e|h)$.

Applies a trade-off between

- Model simplicity
- Data fit

MML Metric



Equivalent to that h that maximizes $P(h)P(e|h)$ — i.e., $P(h|e)$.

$$\begin{aligned} I(h,e) &= I(h) + I(e|h) \\ -\log P(h,e) &= -\log P(h) - \log P(e|h) \\ -\log P(h,e) &= -\log P(h)P(e|h) \\ P(h,e) &= P(h)P(e|h) \end{aligned}$$

Hence, $\min I(h,e) \equiv \max P(h,e)$.

MML Metric for Linear Models

- Network:

$$\log n! + \frac{n(n-1)}{2} - \log E$$

  - $\log n!$ for variable order
  - $\frac{n(n-1)}{2}$ for connectivity
  - $-\log E$ restore efficiency by subtracting cost of selecting a linear extension

- Parameters given dag h:

$$\sum_{X_j} -\log \frac{f(\theta_j|h)}{\sqrt{F(\theta_j)}}$$

  where θ_j are the parameters for X_j and $F(\theta_j)$ is the Fisher information. $f(\theta_j|h)$ is assumed to be $N(0, \sigma_j)$.

  (Cf. with MDL's fixed length for parms)

MML Metric for Linear Models

- Sample for X_j given h and θ_j:

$$-\log P(e|h,\theta_j) = \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi\sigma_j}} e^{-\epsilon_{jk}^2/2\sigma_j^2}$$

  where K is the number of sample values and ϵ_{jk} is the difference between the observed value of X_j and its linear prediction.

MML Metric for discrete models

We can use $P_{CH}(h_i, e)$ (from Cooper & Herskovits) to define an MML metric for discrete models.

Difference between MML and Bayesian metrics:

> MML partitions the parameter space and selects optimal parameters.

Equivalent to a penalty of $\frac{1}{2} \log \frac{\pi e}{6}$ per parameter (Wallace & Freeman 1987); hence:

$$I(e, h_i) = \frac{s_j}{2} \log \frac{\pi e}{6} - \log P_{CH}(h_i, e) \tag{2}$$

Applied in MML Sampling algorithm.

MML search algorithms

MML metrics need to be combined with search. This has been done three ways:

1. Wallace, Korb, Dai (1996): greedy search (linear).
   - Brute force computation of linear extensions (small models only).
2. Neil and Korb (1999): genetic algorithms (linear).
   - Asymptotic estimator of linear extensions
   - GA chromosomes = causal models
   - Genetic operators manipulate them
   - Selection pressure is based on MML
3. Wallace and Korb (1999): MML sampling (linear, discrete).
   - Stochastic sampling through space of totally ordered causal models
   - No counting of linear extensions required

MML Sampling

Search space of totally ordered models (TOMs).
Sampled via a Metropolis algorithm (Metropolis et al. 1953).
From current model M, find the next model M' by:

- Randomly select a variable; attempt to swap order with its predecessor.
- Or, randomly select a pair; attempt to add/delete an arc.

Attempts succeed whenever $P(M')/P(M) > U$ (per MML metric), where U is uniformly random from $[0:1]$.

MML Sampling

Metropolis: this procedure samples TOMs with a frequency proportional to their posterior probability.

To find posterior of dag h: keep count of visits to all TOMs consistent with h

> Estimated by counting visits to all TOMs with identical max likelihoods to h

Output: Probabilities of

- Top dags
- Top statistical equivalence classes
- Top MML equivalence classes

Empirical Results

A weakness in this area — and AI generally.

- Paper publications based upon very small models, loose comparisons.
- ALARM net often used — everything gets it to within 1 or 2 arcs.

Neil and Korb (1999) compared CaMML and BGe (Heckerman & Geiger's Bayesian metric over equivalence classes), using identical GA search over linear models:

- On KL distance and topological distance from the true model, CaMML and BGe performed nearly the same.
- On test prediction accuracy on *strict effect nodes* (those with no children), CaMML clearly outperformed BGe.

KEBN: Overview

- The BN Knowledge Engineering Process
- Model construction
  - Variables and values
  - Graph Structure
  - Probabilities
  - Preferences
- Evaluation

Knowledge Engineering with Bayesian Networks (KEBN)

(Laskey, 1999).

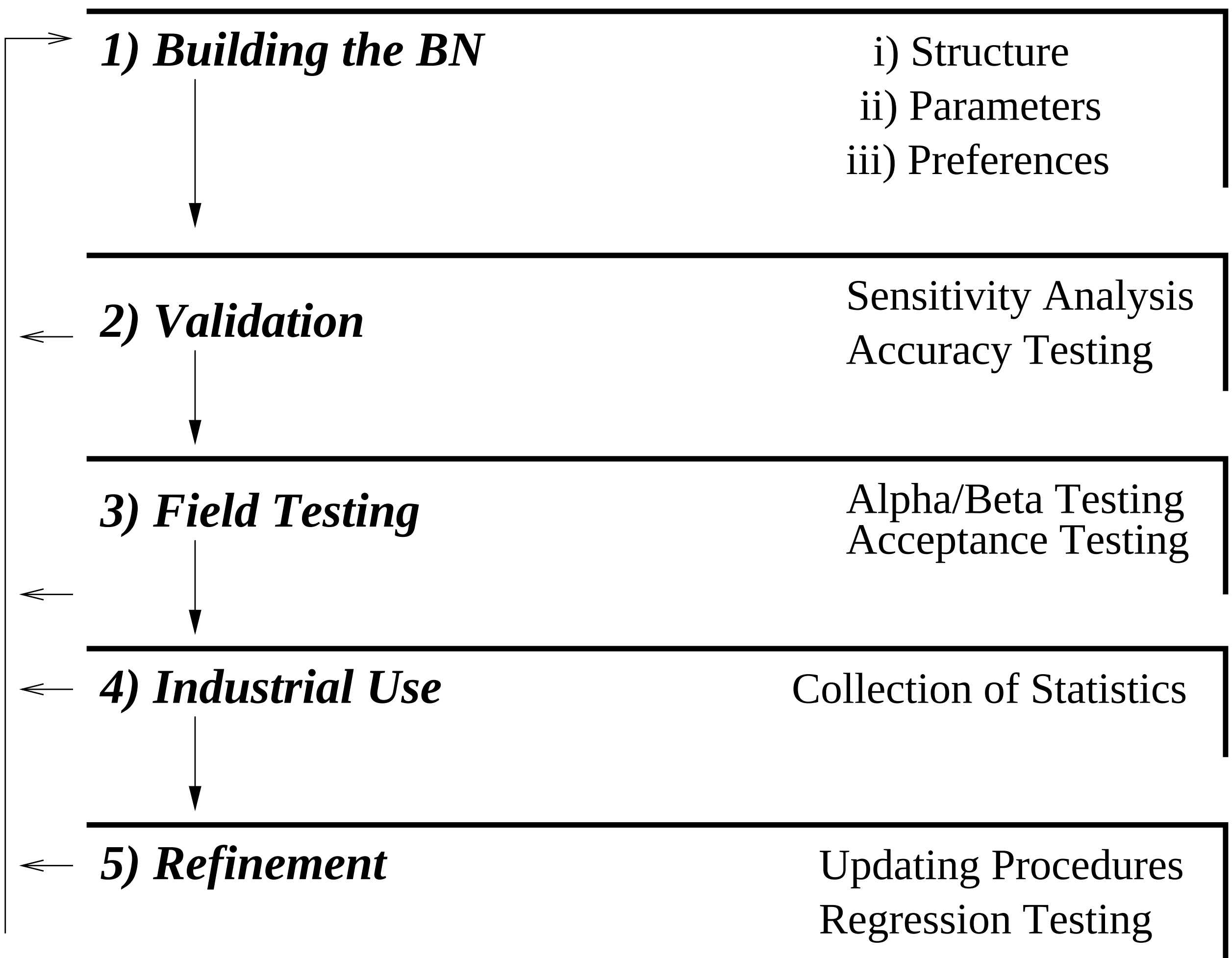
- Objective: Construct a model to perform a defined task
- Participants: Collaboration between domain expert(s) and BN modelling expert(s), including use of automated methods.
- Process: iterate until "done"
  - Define task objective
  - Construct model
  - Evaluate model

KEBN

Production of Bayesian/decision nets for

- **Decision making:** Which policy carries the least risk of failure?
- **Forward Prediction:** Hypothetical or factual. Who will win the election?
- **Retrodiction/Diagnosis:** Which illness do these symptoms indicate?
- **Monitoring/control:** Do containment rods needs to be inserted here at Chernobal?
- **Explanation:** Why did the patient die? Which cause exerts the greater influence?
- **Sensitivity Analysis:** What range of probs/utilities make no difference to X?
- **Information value:** What's the differential utility for changing precision of X to ϵ?

KEBN Lifecycle Model



Notes on Lifecycle Model

- Phase 1: Building Bayesian Networks.
  - Major network components: structure, parameters and utilities.
  - Elicitation: from experts, learned with data mining methods, or some combination of the two.
- Phase 2: Evaluation.
  - Networks need to be validated for: predictive accuracy, respecting known temporal order of the variables and respecting known causal structure.
  - Use statistical data (if available) or expert judgement.
- Phase 3: Field Testing.
  - Domain expert use BN to test usability, performance, etc.

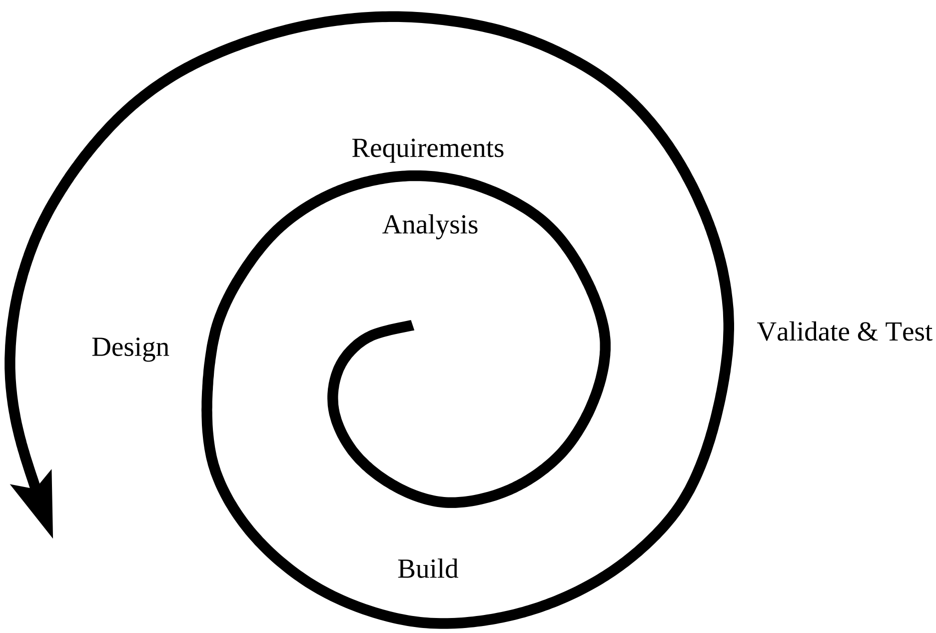
Notes on Lifecycle Model (cont.)

- Phase 4: Industrial Use.
  - Requires a statistics collection regime for on-going validation and/or refinement of the networks.
- Phase 5: Refinement.
  - Requires a process for receiving and incorporating change i requests
  - Includes regression testing to verify that changes do not undermine established performance.

KEBN Spiral Model

From Laskey & Mahoney (2000)

Idea (from Boehm, Brooks): prototype-test cycle



KEBN Tasks

For Bayesian Networks, identifying:

1. What are the variables? What are their values/states?
2. What is the graph structure? What are the direct causal relationships?
3. What are the parameters (probabilities)? Is there local model structure?

When building decision nets, identifying:

4. What are the available actions/decisions?
5. What are the utility nodes & their dependencies?
6. What are the preferences (utilities)?

The major methods are:

- Expert elicitation (1-6)
- Automated learning from data (1-3, 5-6?)
- Adapting from data (1-3, 5-6?)

Variables

Which are the most important variables?

- "Focus" or "query" variables
  - variables of interest
- "Evidence" or "observation" variables
  - What sources of evidence are available?
- "Context" variables
  - Sensing conditions, background causal conditions
- "Controllable" variables
  - variables that can be "set", by intervention

Start with query variables and spread out to related variables.

NB: Roles of variables may change.

Variable values/states

- Variable values must be exclusive and exhaustive
  - Naive modelers sometimes create separate (often Boolean) variables for different states of the same variable
- Types of variables
  - Binary (2-valued, including Boolean)
  - Qualitative
  - Numeric discrete
  - Numeric continuous
- Dealing with infinite and continuous variable domains
  - Some BN software (e.g. Netica) requires that continuous variables be discretized
  - Discretization should be based on differences in effect on related variables (i.e. not just be even sized chunks)

Graphical structure

Goals in specifying graph structure

- Minimize probability elicitation: fewer nodes, fewer arcs, smaller state spaces
- Maximize fidelity of model
  - Sometimes requires more nodes, arcs, states
  - Tradeoff between more accurate model and cost of additional modelling
  - Too much detail can decrease accuracy
- Drawing arcs in causal direction is not "required" BUT
  - Increases conditional independence
  - Results in more compact model
  - Improves ease of probability elicitation
- If mixing continuous and discrete variables
  - Exact inference algorithms only for the case where discrete variables are ancestors, not descendants of continous variables

Relationships between variables

Types of qualitative understanding can help determine local/global structure

- Causal relationships
  - Variables that could cause a variable to take a particular state
  - Variables that could prevent a variable taking a particular state
- Enabling variables
  - Conditions that permit, enhance or inhibit operation of a cause
- Effects of a variable
- Associated variables
  - When does knowing a value provide information about another variable?

Relationships between variables (cont.)

- Dependent and independent variables
  - D-separation tests
  - Which pairs are directly connected? Probabilities dependent regardless of all other variables?

  Matilda - software tool for visual exploration of dependencies (Boneh, 2002)
- Temporal ordering of variables
- Explaining away/undermining
- Causal non-interaction/additivity
- Causal interaction
  - Positive/negative Synergy
  - Preemption
  - Interference/XOR
- Screening off: causal proximity
- Explanatory value
- Predictive value

Probabilities

- The parameters for a BN are a set of conditional probability distributions of child values given values of parents
- One distribution for each combination of values of parent variables
- Assessment is exponential in the number of parent variables
- The number of parameters can be reduced by taking advantage of additional structure in the domain (called **local model structure**)

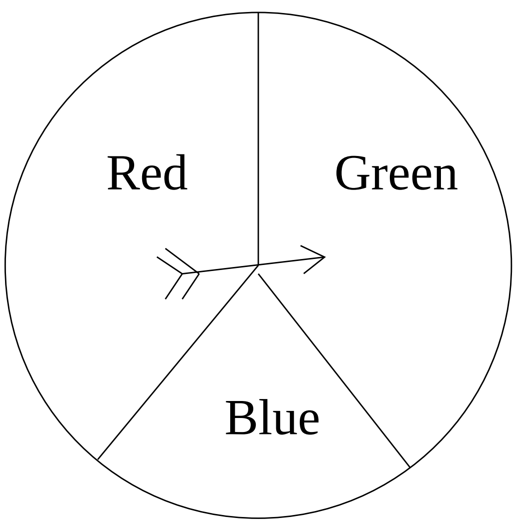
Probability Elicitation

- Discrete variables
  - Direct elicitation: “p=0.7”
  - Odds (esp. for very small probs): “1 in 10,000”
  - Qualitative assessment: “very high probability”
    * Use scale with numerical and verbal anchors (van der Gaag et al., 1999)
    * Do mapping separately from qualitative assessment
- Continuous variables
  - bi-section method
    * Elicit median: equally likely to be above and below
    * Elicty 25th percentile: bisects interval below median
    * Continue with other percentiles till fine enough discriminations
  - Often useful to fit standard functional form to expert’s judgements
  - Need to discreteize for most BN software

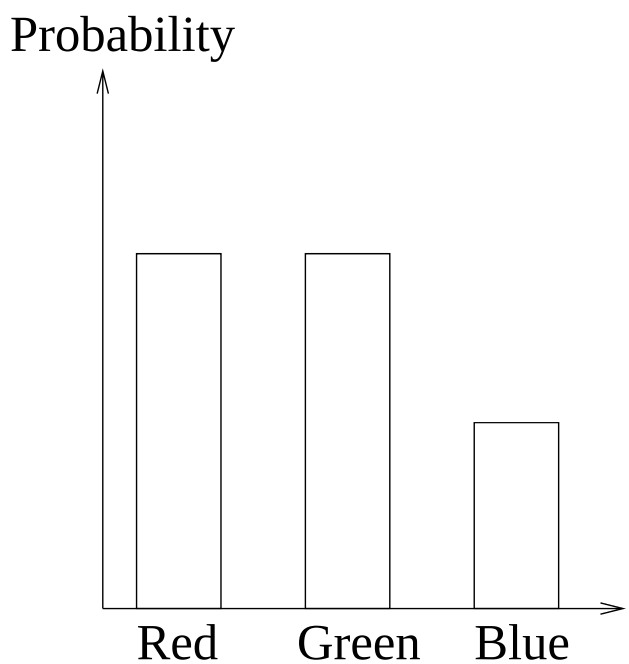
Probability Elicitation

Graphical aids are known to be helpful

- pie charts
- histograms



(a)



(b)

Probability Elicitation (cont.)

- Combination of qualitative and quantitative assessment
- Automated correction of incoherent probabilities (Hope, Korb & Nicholson, 2002)
  - Minimizing squared deviations from original estimates
- Automated maxentropy fill of CPTs (Hope, Korb & Nicholson, 2002)
- Automated normalization of CPTs (Hope, Korb & Nicholson, 2002)
- Use of lotteries to force estimates (also useful for utility elicitation)

Local model structure

Not every cell in CPT is independent from every other cell. Examples:

- Deterministic nodes
  - It is possible to have nodes where the value of a child is exactly specified (logically or numerically) by its parents
- Linear relationships:

$$X_i = a_0 X_0 + \ldots a_n X_n + \epsilon_i$$

- Logit model (binary, 2 parents):

$$\log \frac{P(X_2 | X_0, X_1)}{P(\neg X_2 | X_0, X_1)} = a + bX_0 + cX_1 + dX_1 X_2$$

- Partitions of parent state space
- Independence of causal influence
- Contingent substructures

Elicitation by Partition

(See Heckerman, 1991)

- Partition state set of parents into subsets
  - set of subsets is called a partition
  - each subset is a partition element
- Elicit one probability distribution per partition element
- Child is independent of parent given partition element
- Examples
  - P(reportedLoc|loc, sensor-type,weather) independent of sensor type given weather = sunny
  - P(fever=high|disease) is the same for disease $\in$ {flu,measles}.

Independence of Causal Influence (ICI)

- Assumption: causal influences operate independently of each other in producing effect
  - Probability that C1 causes effect does not depend on whether C2 is operating
  - Excludes synergy or inhibition
- Examples
  - Noisy logic gates (Noisy-OR, Noisy-AND, Noisy-XOR)
  - Noisy adder
  - Noisy max
  - General noisy deterministic function

Noisy-OR nodes

- Adds some uncertainty to logical OR.

  Example: $Fever$ is true if and only if $Cold$, Flu or $Malaria$ is true.

  Assumptions:

  - each cause has an independent chance of causing the effect.
  - all possible causes are listed
  - inhibitors are independent

    E.g.: whatever inhibits $Cold$ from causing $Fever$ is independent of whatever inhibits Flu from causing a $Fever$.

- Inhibitors summarised as “noise parameters”.

Noisy-OR parameters

E.g. if $P(Fever|Cold) = 0.4$, $P(Fever|Flu) = 0.8$, and $P(Fever|Malaria) = 0.9$, then noise parameters are 0.6, 0.2 and 0.1 respectively.

Probability that output node is $False$ is the product of the noise parameters for all the input nodes that are true.

$Cold$	Flu	Mal	$P(Fev)$	$P(\neg Fev)$
F	F	F	0.0	1.0
F	F	T	0.9	0.1
F	T	F	0.8	0.2
F	T	T	0.98	$0.02 = 0.2 \times 0.1$
T	F	F	0.4	0.6
T	F	T	0.94	$0.06 = 0.6 \times 0.1$
T	T	F	0.88	$0.12 = 0.6 \times 0.2$
T	T	T	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

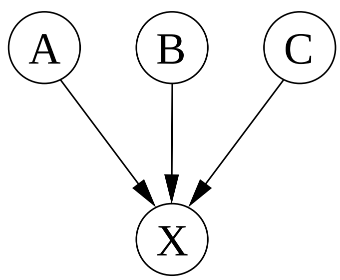
Savings: for binary noisy-OR

CPT requires $2^{10} = 1024$ parameters;
noisy-OR requires 11 parameters

Classification Tree Repn

(Boutillier et al. 1996).

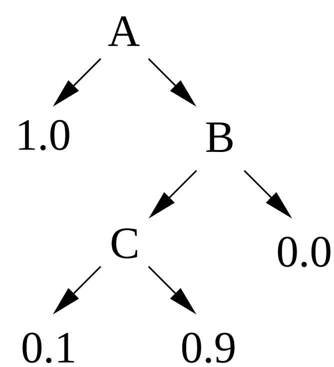
Example: Suppose node X has 3 parents, A, B and C (all nodes Boolean).



(a)

A	B	C	P(X|A,B,C)
T	T	T	1.0
T	T	F	1.0
T	T	T	1.0
T	F	F	1.0
F	T	T	0.1
F	T	F	0.9
F	F	T	0.0
F	F	F	0.0

(b)



(c)

Savings: CPT = 8, tree rep = 4 parameters.

Object-oriented BNs

- Facilitate network construction wrt both structure and probabilities
- Allow representation of commonalities across variables
- Inheritance of priors and CPTs

OOBNs are not supported by the Netica BN software package at all; a version recently in Hugin.

As yet, not widely used.

Decision Analysis

Since 1970s there have been nice software packages for decision analysis:

- Eliciting actions
- Eliciting utilities
- Eliciting probabilities
- Building decision trees
- Sensitivity analysis, etc.

See: Raiffa's *Intro to Decision Analysis* (an excellent book!)

Main differences from KEBN:

- *Scale:* tens vs thousands of parms!!
- *Structure:* trees reflect state-action combinations, not causal structure, prediction, intervention

Eliciting Decision Networks

- Action nodes: What actions can be taken in domain?
- Utility node(s):
  - What unit(s) will "utile" be measured in?
  - Are there difference aspects to the utility that should each be represented in a separate utility node?
- Graph structure:
  - Which variables can decision/actions affects?
  - Does the action/decision affect the utility?
  - What are the outcome variables that there are preferences about?

Model Evaluation

- Elicitation review
  - Review variable and value definition
    * clarity test, agreement on definitions, consistency
  - Review graph and local model structure
  - Review probabilities
    * compare probabilities with each other
- Sensitivity analysis (Laskey, 1993)
  - Measures effect of one variable on another
- Case-based evaluation
  - Run model on test of test cases
  - Compare with expert judgement or "ground truth"
- Validation methods using data (if available)
  - Predictive Accuracy
  - Expected value
  - Kullback-Leibler divergence
  - (Bayesian) Information reward

The need to prototype!

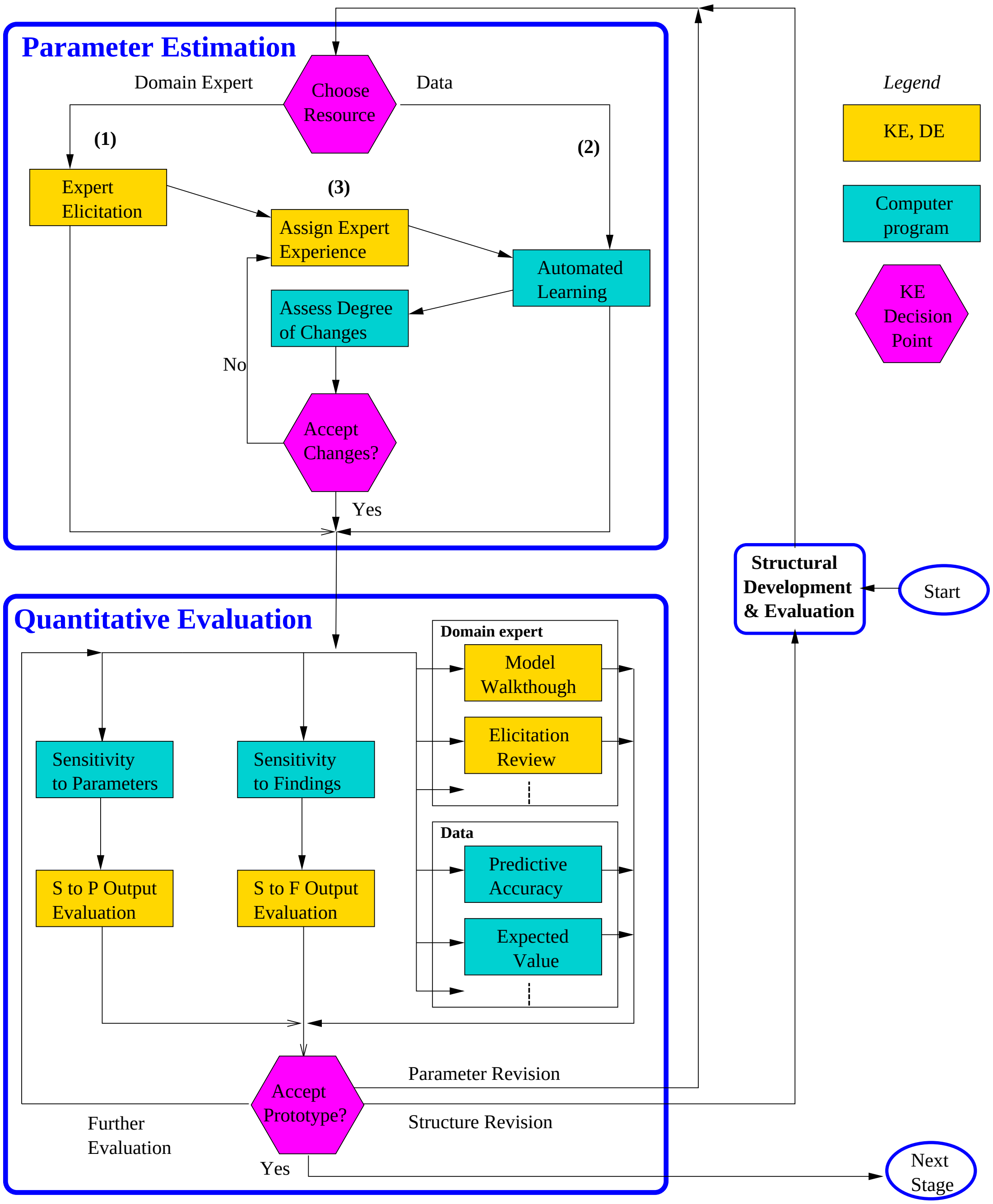
Why prototype?

- It's just the best software development process overall (Brooks). Organic growth of software:
  - tracks the specs
  - has manageable size (at least initially)
- Attacks the comprehensiveness vs. intelligibility trade-off from the right starting point.
- Few off-the-shelf models; prototyping helps us fill in the gaps, helps write the specs

Prototypes

- Initial prototypes minimize risk
  - Don't oversell result
  - Employ available capabilities
  - Simplify variables, structure, questions answered
  - Provide working product for assessment
- Incremental prototypes
  - Simple, quick extension to last
  - Attacks high priority subset of difficult issues
  - Helps refine understanding of requirements/approach

More recent KEBN methodologies



KEBN Summary

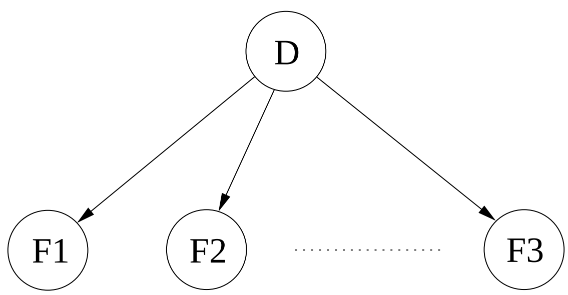
- Various BN structures are available to compactly and accurately represent certain types of domain features.
- There is an interplay between elements of the KE process: variable choice, graph structure and parameters.
- No standard knowledge engineering process exists as yet.
- Integration of expert elicitation and automated methods still in early stages.
- There are few existing tools for supporting the BN KE process.
  - We at Monash are developing some! (e.g. VerbalBN, Matilda)

BN Applications

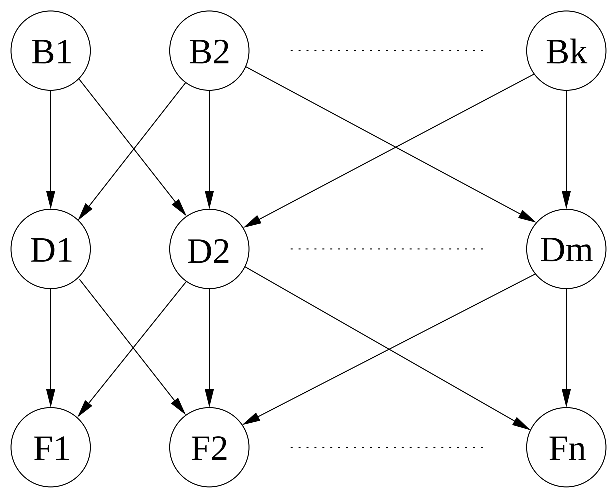
- Most BN applications to date are hand-crafted using domain information provided by experts.
- Tasks include:
  - prediction: (1) given evidence; (2) effect of intervention.
  - diagnosis
  - planning
  - decision making
  - explanation
  - choice of observations (experimental design)

Medical applications: network structure

- Simplest tree-structured network for diagnostic reasoning: H = disease hypothesis, F = findings (symptoms, test results).



(a)

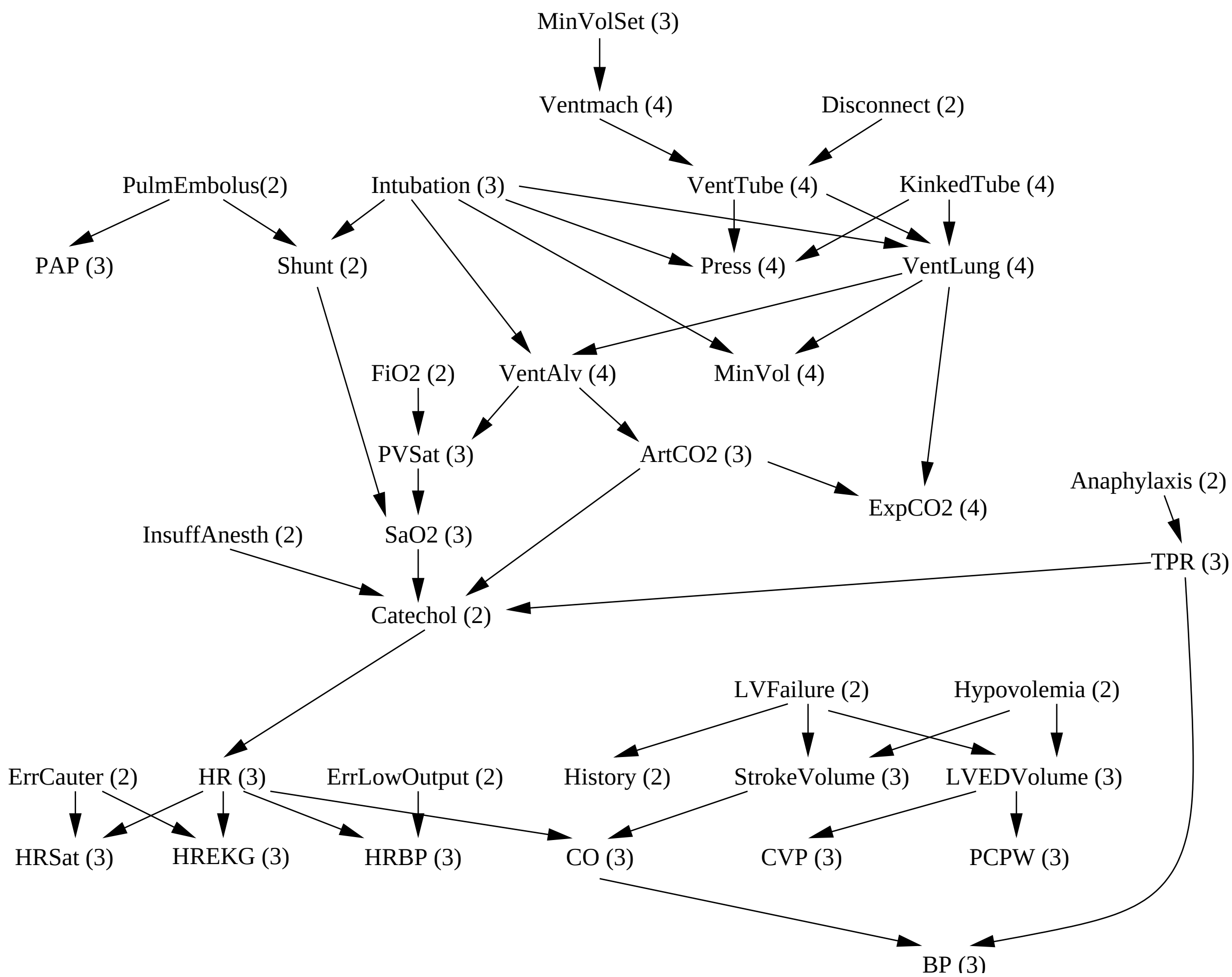


(b)

- Multiply-connected network (QMR structure): B = background information (e.g. age, sex of patient)

The ALARM network

ALARM (Beinlich *et al.*, 1989): 37 nodes, 42 arcs. (Benchmark network often used in literature. See Netica examples.)



Monash BN Applns: Overview

- User modelling (plan recognition in a MUD, web page pre-fetching): Zukerman, Albrecht, Nicholson (1997-2001)
- Ambulation monitoring and fall detection: Nicholson, Brown (Monash Biomedical Engineering), Honours projects 1997, 2000
- Weather forecasting (Bureau of Meteorology)
  - Seabreeze prediction: Kennett, Nicholson, Korb, Ryan, 2001 Honours project
  - BNs for forecasting decision support: Boneh, Nicholson, Korb, Bally 2002-2004 ARC Linkage Project
- Intelligent tutoring for decimal understanding: Nicholson, Boneh, University of Melbourne (1999-2003)
- NAG (Nice Argument Generator): Zukerman, Korb
- Bayesian Poker: Korb, Nicholson, Honours projects 1993,1994,1995,2001,2003

Monash BN Applns (cont)

- SARBayes: Twardy, Korb, Albrecht, Victorian Search and Rescue, 2001 Honours project
- Ecological risk assessment:
  - Nicholson, Korb, Pollino (Monash Centre for Water Studies), 2003-2005 Native Fish abundance in Goulburn Water
  - Predicting recreational water quality: Twardy, Nicholson, NSW EPA, 2003 Honours project
  - Tropical seagrass in great barrier reef: Nicholson, Thomas (Monash Centre for Water Studies), 2004-2006
- Predicting cardiovascular risk from epidemiological data: Korb, Nicholson, Twardy, John McNeil (Department of Epidemiology and Preventive Medicine, Monash University), 2004-2006
- Change impact analysis in software architecture design: Nicholson, Tang, Jin, Han (Swinburne)

References

Introduction to Bayesian AI

T. Bayes (1764) "An Essay Towards Solving a Problem in the Doctrine of Chances." *Phil Trans of the Royal Soc of London*. Reprinted in *Biometrika, 45 (1958),* 296-315.

B. Buchanan and E. Shortliffe (eds.) (1984) *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley.

R. Cox (1946). Probability, frequency and reasonable expectation. *American Journal of Physics, 14,* 1-13, 1946.

B. de Finetti (1964) "Foresight: Its Logical Laws, Its Subjective Sources," in Kyburg and Smokler (eds.) *Studies in Subjective Probability.* NY: Wiley.

G. Gigerenzer and U. Hoffrage (1995). "How to improve Bayesian reasoning without instruction: Frequency formats." *Psychological Review, 102,* 684-704.

A. Hajek (2002). "Scotching Dutch Books." Talk at Monash University, July, 2002.

D. Heckerman (1986) "Probabilistic Interpretations for MYCIN's Certainty Factors," in L.N. Kanal and J.F. Lemmer (eds.) *Uncertainty in Artificial Intelligence*. North-Holland.

E. J. Horvitz, J. S. Breese, and M. Henrion (1988). Decision theory in expert systems and artificial intelligence. *International Journal of Approximate Reasoning, 2,* 247-302.

C. Howson and P. Urbach (1993) *Scientific Reasoning: The Bayesian Approach.* Open Court.
A MODERN REVIEW OF BAYESIAN THEORY.

D. Hume (1737) *A Treatise of Human Nature*.

D. Kahneman, P. Slovic and A. Tversky (eds.) (1982) *Judgment under uncertainty: Heuristics and biases.* Cambridge.

K.B. Korb (1995) “Inductive learning and defeasible inference,” *Jrn for Experimental and Theoretical AI, 7,* 291-324.

R. Neapolitan (1990) *Probabilistic Reasoning in Expert Systems*. Wiley.
CHAPTERS 1, 2 AND 4 COVER SOME OF THE RELEVANT HISTORY.

J. Pearl (1988) *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann.

F. P. Ramsey (1926/1931) “Truth and Probability” in *The Foundations of Mathematics and Other Essays.* NY: Humanities Press.
THE ORIGIN OF MODERN BAYESIANISM. INCLUDES LOTTERY-BASED ELICITATION AND DUTCH-BOOK ARGUMENTS FOR THE USE OF PROBABILITIES.

R. Reiter (1980) "A logic for default reasoning," *Artificial Intelligence, 13,* 81-132.

J. von Neumann and O.Morgenstern (1947) *Theory of Games and Economic Behavior,* 2nd ed. Princeton Univ.
Standard reference on eliciting utilities via lotteries.

Bayesian Networks

E. Charniak (1991) "Bayesian Networks Without Tears", *Artificial Intelligence Magazine*, pp. 50-63, Vol 12.
An elementary introduction.

G.F. Cooper (1990) The computational complexity of probabilistic inference using belief networks. *Artificial Intelligence, 42,* 393-405.

R. G. Cowell, A. Philip Dawid, S. L. Lauritzen and D. J. Spiegelhalter (1999) *Probabilistic networks and expert systems.* New York: Springer.
technical survey of Bayesian net technology, including learning Bayesian nets.

D. D'Ambrosio (1999) "Inference in Bayesian Networks". *Artificial Intelligence Magazine*, Vol 20, No. 2.

A. P. Dawid (1998) Conditional independence. In *Encyclopedia of Statistical Sciences, Update Volume 2.* New York: Wiley Interscience.

P. Haddaway (1999) "An Overview of Some Recent Developments in Bayesian Problem-Solving Techniques". *Artificial Intelligence Magazine*, Vol 20, No. 2.

R.A. Howard & J.E. Matheson (1981) Influence Diagrams. In Howard and Matheson (eds.) *Readings in the Principles and Applications of Decision Analysis*. Menlo Park, Calif: Strategic Decisions Group.

F. V. Jensen (1996) *An Introduction to Bayesian Networks*, Springer.

K.B. Korb, L.R. Hope, A.E. Nicholson and K. Axnick. Varieties of Causal Intervention In C. Zhang, H.W. Guesgen and W.K. Yeap (eds), *Lecture Notes in Artificial Intelligence, (Proceedings of the 8th Pacific Rim International Conference on Artificial Intelligence,[PRICAI 2004], Auckland, New Zealand, August 2004)*, Springer-Verlag, Berlin, Germany, Vol 3157, pp 322-331.

R. Neapolitan (1990) *Probabilistic Reasoning in Expert Systems*. Wiley.
Similar coverage to that of Pearl; more emphasis on practical algorithms for network updating.

J. Pearl (1988) *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann.
This is the classic text introducing Bayesian networks to the AI community.

J. Pearl (2000) *Causality.* Cambridge University.

Poole, D., Mackworth, A., and Goebel, R. (1998) *Computational Intelligence: a logical approach*. Oxford University Press.

Russell & Norvig (1995) *Artificial Intelligence: A Modern Approach*, Prentice Hall.

J. Whittaker (1990) *Graphical models in applied multivariate statistics.* Wiley.

Learning Bayesian Networks

H. Blalock (1964) *Causal Inference in Nonexperimental Research.* University of North Carolina.

R. Bouckeart (1994) *Probabilistic network construction using the minimum description length principle.* Technical Report RUU-CS-94-27, Dept of Computer Science, Utrecht University.

C. Boutillier, N. Friedman, M. Goldszmidt, D. Koller (1996) "Context-specific independence in Bayesian networks," in Horvitz & Jensen (eds.) *UAI 1996*, 115-123.

G. Brightwell and P. Winkler (1990) *Counting linear extensions is #P-complete.* Technical Report DIMACS 90-49, Dept of Computer Science, Rutgers Univ.

W. Buntine (1991) "Theory refinement on Bayesian networks," in D'Ambrosio, Smets and Bonissone (eds.) *UAI 1991,* 52-69.

W. Buntine (1996) "A Guide to the Literature on Learning Probabilistic Networks from Data," *IEEE Transactions on Knowledge and Data Engineering,8,* 195-210.

D.M. Chickering (1995) “A Tranformational Characterization of Equivalent Bayesian Network Structures,” in P. Besnard and S. Hanks (eds.) *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence* (pp. 87-98). San Francisco: Morgan Kaufmann.

STATISTICAL EQUIVALENCE.

Chickering, D. M. (2002). “Optimal structure idenification with greedy search.” *Journal of Machine Leaning Research, 3,* 507-559.

G.F. Cooper and E. Herskovits (1991) “A Bayesian Method for Constructing Bayesian Belief Networks from Databases,” in D’Ambrosio, Smets and Bonissone (eds.) *UAI 1991,* 86-94.

G.F. Cooper and E. Herskovits (1992) “A Bayesian Method for the Induction of Probabilistic Networks from Data,” *Machine Learning, 9,* 309-347.

AN EARLY BAYESIAN CAUSAL DISCOVERY METHOD.

H. Dai, K.B. Korb, C.S. Wallace and X. Wu (1997) “A study of casual discovery with weak links and small samples.” *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI),* pp. 1304-1309. Morgan Kaufmann.

N. Friedman (1997) “The Bayesian Structural EM Algorithm,” in D. Geiger and P.P. Shenoy (eds.) *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence* (pp. 129-138). San Francisco: Morgan Kaufmann.

D. Geiger and D. Heckerman (1994) "Learning Gaussian networks," in Lopes de Mantras and Poole (eds.) *UAI 1994,* 235-243.

D. Heckerman and D. Geiger (1995) "Learning Bayesian networks: A unification for discrete and Gaussian domains," in Besnard and Hanks (eds.) *UAI 1995,* 274-284.

D. Heckerman, D. Geiger, and D.M. Chickering (1995) "Learning Bayesian Networks: The Combination of Knowledge and Statistical Data," *Machine Learning, 20,* 197-243.
BAYESIAN LEARNING OF STATISTICAL EQUIVALENCE CLASSES.

K. Korb (1999) "Probabilistic Causal Structure" in H. Sankey (ed.) *Causation and Laws of Nature: Australasian Studies in History and Philosophy of Science 14*. Kluwer Academic.
INTRODUCTION TO THE RELEVANT PHILOSOPHY OF CAUSATION FOR LEARNING BAYESIAN NETWORKS.

P. Krause (1998) *Learning Probabilistic Networks.*
`http : //www.auai.org/bayes_USKrause.ps.gz`
BASIC INTRODUCTION TO BNS, PARAMETERIZATION AND LEARNING CAUSAL STRUCTURE.

W. Lam and F. Bacchus (1993) "Learning Bayesian belief networks: An approach based on the MDL principle," *Jrn Comp Intelligence, 10,* 269-293.

D. Madigan, S.A. Andersson, M.D. Perlman & C.T. Volinsky (1996) "Bayesian model averaging and model selection for Markov equivalence classes of acyclic digraphs," *Comm in Statistics: Theory and Methods, 25,* 2493-2519.

D. Madigan and A. E. Raftery (1994) "Model selection and accounting for model uncertainty in graphical modesl using Occam's window," *Jrn AMer Stat Assoc, 89,* 1535-1546.

Meek, C. (1996). *Graphical models: Selectiong causal and statistical models.* PhD disseration, Philosophy, Carnegie Mellon University.

N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller (1953) "Equations of state calculations by fast computing machines," *Jrn Chemical Physics, 21,* 1087-1091.

J.R. Neil and K.B. Korb (1999) "The Evolution of Causal Models: A Comparison of Bayesian Metrics and Structure Priors," in N. Zhong and L. Zhous (eds.) *Methodologies for Knowledge Discovery and Data Mining: Third Pacific-Asia Conference* (pp. 432-437). Springer Verlag.
GENETIC ALGORITHMS FOR CAUSAL DISCOVERY; STRUCTURE PRIORS.

J.R. Neil, C.S. Wallace and K.B. Korb (1999) "Learning Bayesian networks with restricted causal interactions," in Laskey and Prade (eds.) *UAI 99,* 486-493.

J. Rissanen (1978) “Modeling by shortest data description,” *Automatica, 14,* 465-471.

R.W. Robinson (1977). “Counting unlabelled acyclic diagraphs,” in C.H.C.Little (Ed.), Lecture notes in mathematics 62: Combinatorial mathematics V. New York: Springer-Verlag.

H. Simon (1954) “Spurious Correlation: A Causal Interpretation,” *Jrn Amer Stat Assoc, 49,* 467-479.

D. Spiegelhalter & S. Lauritzen (1990) “Sequential Updating of Conditional Probabilities on Directed Graphical Structures,” *Networks, 20,* 579-605.

P. Spirtes, C. Glymour and R. Scheines (1990) “Causality from Probability,” in J.E. Tiles, G.T. McKee and G.C. Dean *Evolving Knowledge in Natural Science and Artificial Intelligence.* London: Pitman. AN ELEMENTARY INTRODUCTION TO STRUCTURE LEARNING VIA CONDITIONAL INDEPENDENCE.

P. Spirtes, C. Glymour and R. Scheines (1993) *Causation, Prediction and Search: Lecture Notes in Statistics 81.* Springer Verlag.
A THOROUGH PRESENTATION OF THE ORTHODOX STATISTICAL APPROACH TO LEARNING CAUSAL STRUCTURE.

J. Suzuki (1996) “Learning Bayesian Belief Networks Based on the Minimum Description Length Principle,” in L. Saitta (ed.) *Proceedings of the Thirteenth International Conference on Machine Learning* (pp. 462-470). San Francisco: Morgan Kaufmann.

T.S. Verma and J. Pearl (1991) "Equivalence and Synthesis of Causal Models," in P. Bonissone, M. Henrion, L. Kanal and J.F. Lemmer (eds) *Uncertainty in Artificial Intelligence 6* (pp. 255-268). Elsevier.
THE GRAPHICAL CRITERION FOR STATISTICAL EQUIVALENCE.

C.S. Wallace and D. Boulton (1968) "An information measure for classification," *Computer Jrn, 11,* 185-194.

C.S. Wallace and P.R. Freeman (1987) "Estimation and inference by compact coding," *Jrn Royal Stat Soc (Series B), 49,* 240-252.

C. S. Wallace and K. B. Korb (1999) "Learning Linear Causal Models by MML Sampling," in A. Gammerman (ed.) *Causal Models and Intelligent Data Management.* Springer Verlag.
SAMPLING APPROACH TO LEARNING CAUSAL MODELS; DISCUSSION OF STRUCTURE PRIORS.

C. S. Wallace, K. B. Korb, and H. Dai (1996) "Causal Discovery via MML," in L. Saitta (ed.) *Proceedings of the Thirteenth International Conference on Machine Learning* (pp. 516-524). San Francisco: Morgan Kaufmann.
INTRODUCES AN MML METRIC FOR CAUSAL MODELS.

S. Wright (1921) "Correlation and Causation," *Jrn Agricultural Research, 20,* 557-585.

S. Wright (1934) "The Method of Path Coefficients," *Annals of Mathematical Statistics, 5,* 161-215.

BN Knowledge Engineering

Boneh, T. (2002) “Visualisation of structural dependencies for Bayesian Network Knowledge Engineering”, Masters thesis, University of Melbourne.

C. Boutillier, N. Friedman, M. Goldszmidt, D. Koller (1996) “Context-specific independence in Bayesian networks,” in Horvitz & Jensen (eds.) *UAI 1996*, 115-123.

Druzdzel, M.J. and van der Gaag, L.C. (1995), “Elicitation of probabilities for belief networks: Combining qualitative and quantitative information”, Besnard & Hanks (eds) *UAI95*, pp. 141-148.

M.J. Druzsdel and L.C. van der Gaag (Eds.) (2000) “Building probabilistic networks: Where do the numbers come from?” Editors Introduction to Special Section *IEEE Trans. on Knowledge and Data Engineering*, 12(4), pp 481-486.

D. Heckerman (1990), “Probabilistic Similarity Networks”,, *Networks*, 20, pp. 607-636.

Laskey, K.B. (1993) “Sensitivity Analysis for Probability Assessments in Bayesian Networks”, in Heckerman & Mamdani (eds.) *UAI93*, pp. 136-142.

Laskey, K.B. and Mahoney, S.M. (1997) “Network Fragments: Representing Knowledge for Constructing Probabilistic Models”, in Geiger & Shenoy (eds.) *UAI97*, pp. 334-341.

Laskey, K.B. (1999) A Full-day tutorial given at *UAI99*.
`http://ite.gmu.edu/ klaskey/papers/uai99tutorial.pp`

Laskey, K.B. and Mahoney, S.M. (2000) "Network Engineering for Agile Belief Network Models", *IEEE: Transactions on Knowledge and Data Engineering*, 12(4), pp. 487-498.

Mahoney, S.M. and Laskey, K.B. (1996) "Network Engineering for Complex Belief Networks", in Horvitz & Jensen (eds.) *UAI96*, pp. 389-396.

Monti, S. and Carenini, G. (2000) "Dealing with the Expert Inconsistency in Probability Elicitation", *IEEE: Transactions on Knowledge and Data Engineering*, 12(4), pp. 499-508.

Nikovski, D. (2000) "Constructing Bayesian Networks for Medical Diagnosis from Incomplete and Partially Correct Statistics", *IEEE: Transactions on Knowledge and Data Engineering*, 12(4), pp. 509-516.

M. Pradham, G. Provan, B. Middleton and M. Henrion (1994) "Knowledge engineering for large belief networks", In de Mantras & Poole (eds.) *UAI94*, pp. 484-490.

S. Srinivas (1993) "A Generalization of the Noisy-OR model", In Heckerman & Mamdani (eds.) *UAI93* pp. 208-215.

L.C. van der Gaag, S. Renooij, C.L.M. Witteman, B.M.P. Aleman, B.G. Taal (1999) "How to Elicit Many Probabilities", Laskey & Prade (eds) *UAI99*, pp. 647-654.

O. Woodberry, A.E. Nicholson, K.B. Korb, and C. Pollino (2004). Parameterising Bayesian Networks. in G. I. Webb and X. Yu (eds). *Lecture Notes in Artificial Intelligence, (Proceedings of the 17th Australian Joint Conference on Advances in Artificial Intelligence [AI'04], Cairns, Australia, 4-6 December 2004)*, Springer-Verlag, Berlin, Germany, Vol 3339, pp 1101-1107.

Monash BN Applications

D. Albrecht, I. Zukerman and A. Nicholson (1998) Bayesian Models for Keyhole Plan Recognition in an Adventure Game. In *User Modeling and User-Adapted Interaction*, Kluwer Academic Publishers, Special Issue on Machine Learning in User Modeling, 8(1-2), pp. 5-47, 1998.

T. Boneh, A. Nicholson, and L. Sonenberg (To appear). Matilda: A Visual Tool for Modelling with Bayesian Networks. *International Journal of Intelligent Systems*, Wiley.

J. Carlton (2000) "Bayesian Poker", Honours thesis, School of Computer Science and Software Engineering. www.csse.monash.edu.au/hons/projects/2000/Jason.Carlton/

R.J. Kennett, K.B. Korb, A.E. Nicholson (2001). "Seabreeze Prediction Using Bayesian Networks", In *PAKDD'01 – Proceedings of the Fourth Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Hong Kong, pages 148-153.

K. B. Korb, A. E. Nicholson and N. Jitnah (1999), ”Bayesian poker”, In *UAI’99 - Proceedings of the 15th International Conference on Uncertainty in Artificial Intelligence*, Sweden, pp. 343-350.

A. E. Nicholson and J. M. Brady (1992a) The data association problem when monitoring robot vehicles using dynamic belief networks. In *Proc. of the 10th European Conf. on Artificial Intelligence (ECAI-92)*, pages 689–693, 1992.

A. E. Nicholson and J. M. Brady (1992b). Sensor validation using dynamic belief networks. In *Proceedings of the 8th Conference on Uncertainty in Artificial Intelligence*, pages 207–214, 1992.

A. E. Nicholson and J. M. Brady (1994) Dynamic belief networks for discrete monitoring. *IEEE Systems, Man and Cybernetics*, 24(11):1593–1610, 1994.

A.E. Nicholson (1996) Fall diagnosis using dynamic belief networks. In Foo, N. and Goebel, R. (Eds.), *PRICAI-96 Proceedings – the 4th Pacific Rim International Conf. on Artificial Intelligence (PRICAI-96)*, pp. 206-217, Cairns, Springer: Lecture Notes in Artificial Intelligence, 1996.

A.E. Nicholson, T. Boneh, T. Wilkin, K. Stacey, L. Sonenberg, V. Steinle (2001). “A Case Study in Knowledge Discovery and Elicitation in an Intelligent Tutoring Application”, In *UAI01 – Proc. of the 17th Conference on Uncertainty in Artificial Intelligence* Seattle, 2001, pp. 386-394.

C.A. Pollino, O. Woodberry, A.E. Nicholson, and K.B. Korb. Parameterising Bayesian Networks: A Case Study in Ecological Risk Assessment. In: V. Kachitvichyysnukul, U. Purintrapiban and P. Utayopas (Editors), *Proceedings of the 2005 Conference on Simulation and Modelling*, Bangkok, Thailand, pp. 289-297.

K. Stacey, L. Sonenberg, A.E. Nicholson, T. Boneh, V. Steinle (2003). “A Teaching Model Exploiting Cognitive Conflict Driven by a Bayesian Network”, in P Brusilovsky, A Corbett and F de Rosis (eds), *UM03 – Proc. of the 9th International Conference on User Modeling*, Lecture Notes in Artificial Intelligence, Springer-Verlag, Berlin, Germany, Vol 2702, pp 352-362.

A. Tang, Y. Jin, J. Han, A. Nicholson. Predicting Change Impact in Architecture Design with Bayesian Belief Networks, In Proceedings of WICSA 2005, the 5th Working IEEE/IFIP Conference on Software Architecture, Pittsburgh, Pennsylvannia, USA, November 6-10, 2005.

A. Tang, A. Nicholson, Y. Jin, J. Han (To appear). Using Bayesian Belief Networks for Change Impact Analysis in Architecture Design. *Journal of Systems and Software*.

Thomas, C., B. Hart, A. Nicholson, M. Grace, J. Brodie and C. Pollino (2005). Development of Criteria for Simplifying Ecological Risk Models. In Zerger, A. and Argent, R.M. (eds) MODSIM 2005 International Congress on Modelling and Simulation. Modelling and

Simulation Society of Australia and New Zealand, December 2005, pp. 772-778. ISBN: 0-9758400-2-9. http://www.mssanz.org.au/modsim05/papers/thomas.pdf

C.R. Twardy, A.E. Nicholson, K.B. Korb, J. McNeil (2006). Epidemiological data mining of cardiovascular Bayesian networks. *electronic Journal of Health Informatics (eJHI)*, Vol 1, No 1.

C.R. Twardy, A.E. Nicholson, K.B. Korb and J. McNeil. Data Mining Cardiovascular Bayesian Networks. Technical report 2004/165. School of Computer Science and Software Engineering, Monash University, 2004.

C.R. Twardy, A.E. Nicholson and K.B. Korb: Knowledge Engineering Cardiovascular Bayesian Networks from the Literature, Clayton School of It, Monash University, Melbourne. Technical report 2005/170.

O. Woodberry, A.E. Nicholson, K.B. Korb, and C. Pollino. Parameterising Bayesian Networks. In G I Webb and X Yu (eds). *Lecture Notes in Artificial Intelligence, (Proceedings of the 17th Australian Joint Conference on Advances in Artificial Intelligence [AI'04], Cairns, Australia, 4-6 December 2004)*, Springer-Verlag, Berlin, Germany, Vol 3339, pp 1101-1107.