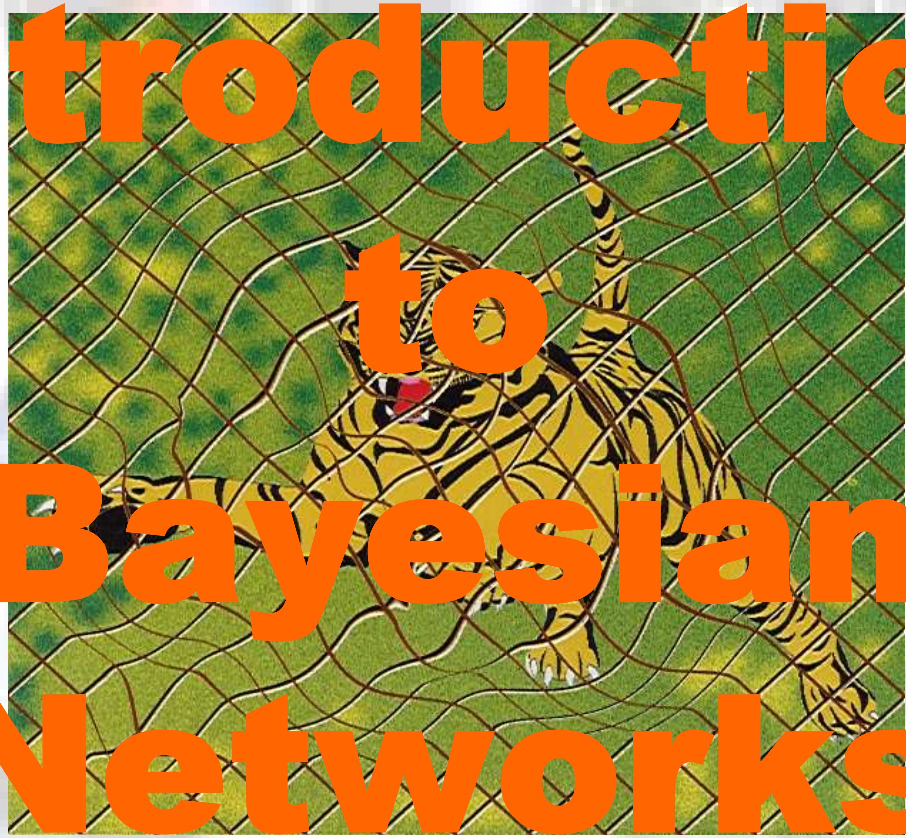
Introduction to Bayesian Networks



On learning and inference

- n binary random variables $X_1,\ldots,X_n$
- A joint probability distribution $P(X_1,\ldots,X_n)$
- Inference:
  - compute the conditional probability distribution for the thing you want to know, given all that you know, marginalizing out all that you don't know and don't want to know
  - In pricinple exponential, requires $O(2^n)$ operations
  - Can be simplified if the joint distribution factorizes by indepencence: $P(A,B)=P(A)P(B)$
- Learning:
  - learn the model structure: what is (conditionally) independent of what
  - learn the parameters defining the ”local” distributions
- Supervised learning: construct directly a model for the required conditional distribution, without forming the joint distribution first

Probabilistic reasoning

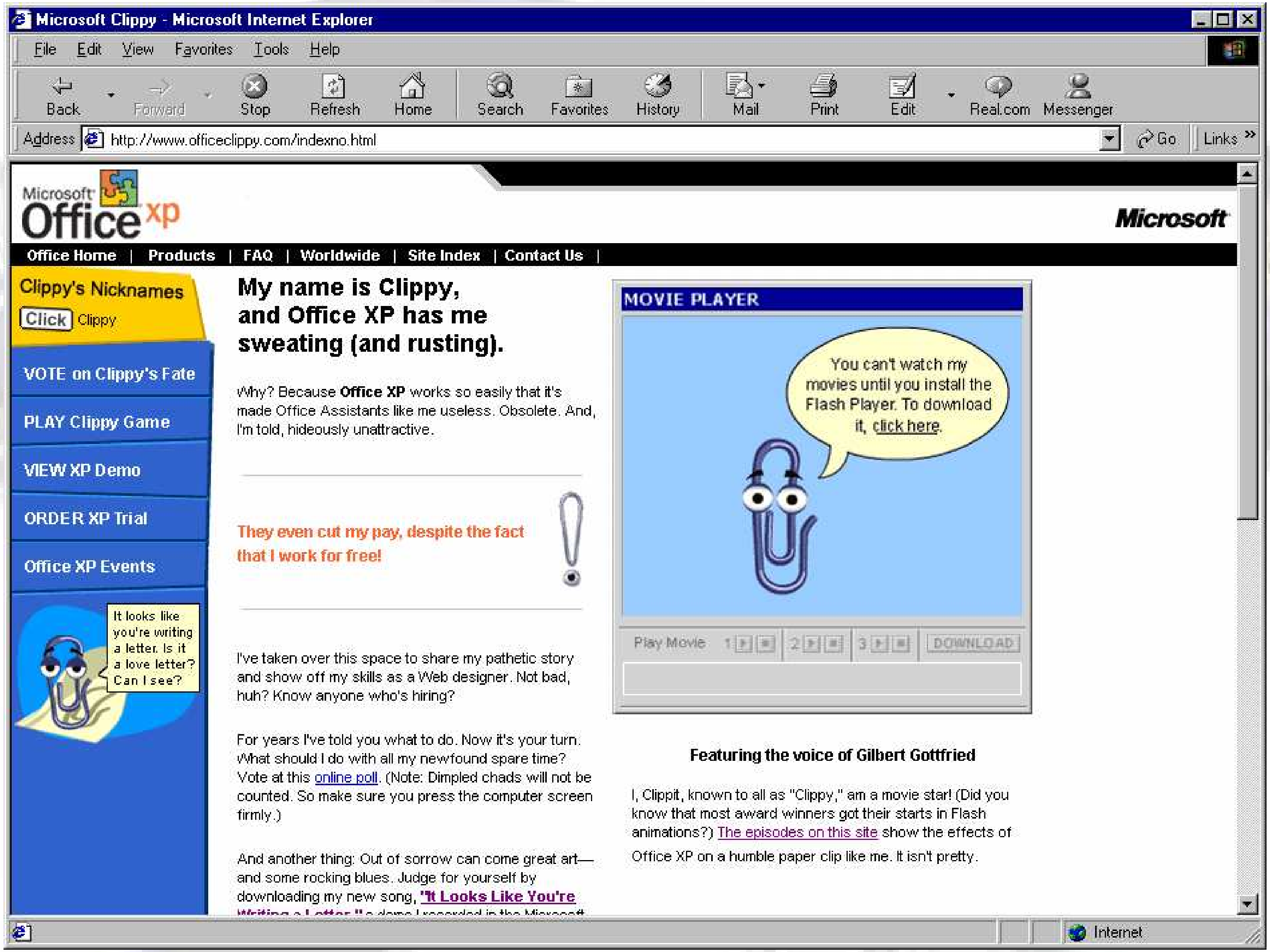
- n (discrete) random variables $X_1,\ldots,X_n$
- joint probability distribution $P(X_1,\ldots,X_n)$
- Input: a partial value assignment Ω,

  $\Omega = < X_1, X_2=x_2, X_3, X_4=x_4, X_5=x_5, X_6,\ldots,X_n >$
- Probabilistic reasoning:
  - compute $P(X=x \mid \Omega)$ for all X not instantiated in Ω, and for all values of X (marginal distribution).
  - find a MAP (maximum a posterior probability) assignment consistent with Ω
- Bayesian networks: a family of probabilistic models and algorithms enabling computationally efficient probabilistic reasoning

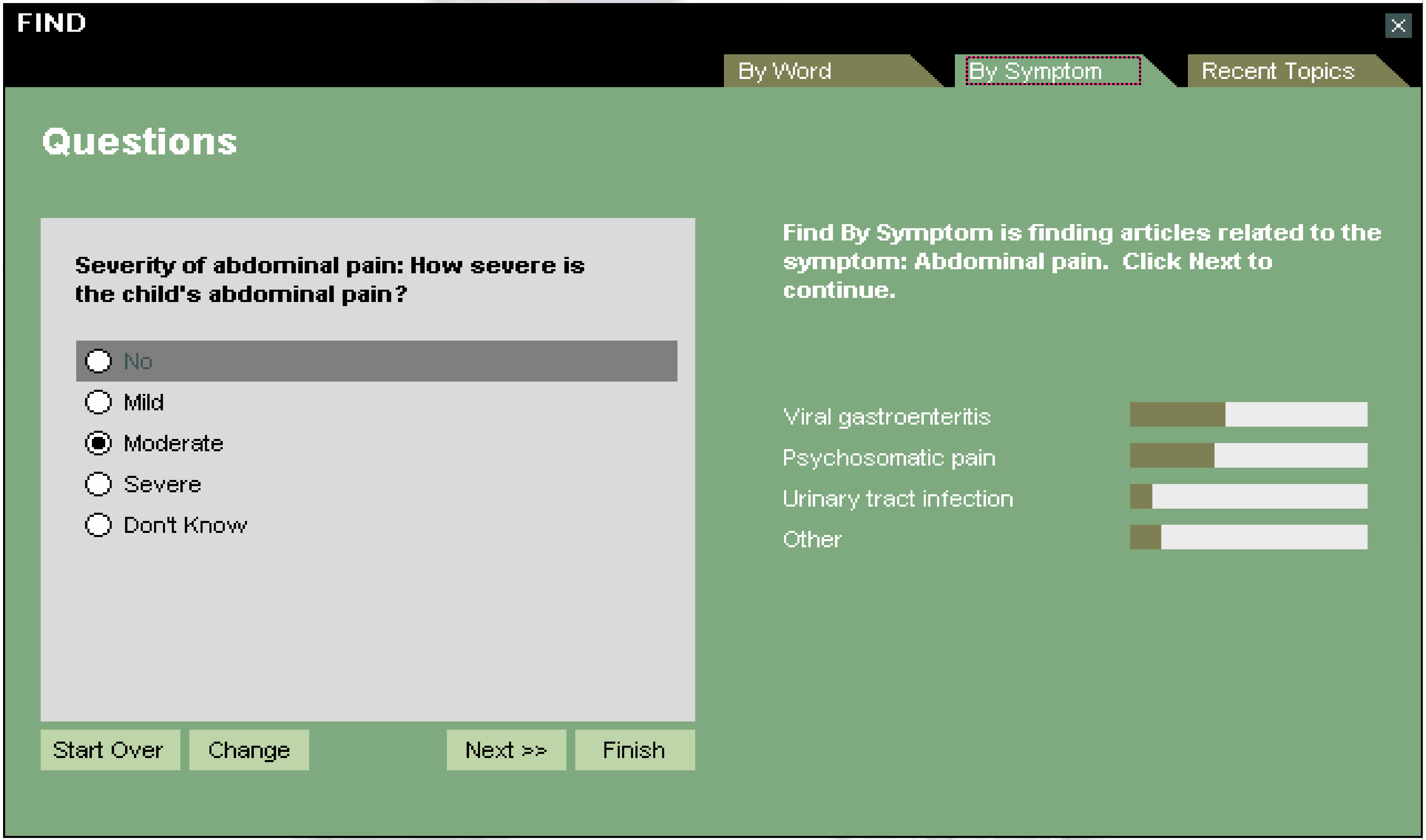
Bayesian networks: a billion dollar perspective

"Microsoft's competitive advantage, he [Gates] responded, was its expertise in "Bayesian networks". Ask any other software executive about anything "Bayesian" and you're liable to get a blank stare. Is Gates onto something? Is this alien-sounding technology Microsoft's new secret weapon?"

(Leslie Helms, Los Angeles Times, October 28, 1996.)


Microsoft Clippy - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Real.com Messenger
Address http://www.officeclippy.com/indexno.html
Go Links
Microsoft Office xp
Microsoft
Office Home | Products | FAQ | Worldwide | Site Index | Contact Us |
Clippy's Nicknames
Click Clippy
VOTE on Clippy's Fate
PLAY Clippy Game
VIEW XP Demo
ORDER XP Trial
Office XP Events
It looks like you're writing a letter. Is it a love letter? Can I see?
My name is Clippy, and Office XP has me sweating (and rusting).
Why? Because Office XP works so easily that it's made Office Assistants like me useless. Obsolete. And, I'm told, hideously unattractive.
They even cut my pay, despite the fact that I work for free!
I've taken over this space to share my pathetic story and show off my skills as a Web designer. Not bad, huh? Know anyone who's hiring?
For years I've told you what to do. Now it's your turn. What should I do with all my newfound spare time? Vote at this online poll. (Note: Dimpled chads will not be counted. So make sure you press the computer screen firmly.)
And another thing: Out of sorrow can come great art—and some rocking blues. Judge for yourself by downloading my new song, "It Looks Like You're
MOVIE PLAYER
You can't watch my movies until you install the Flash Player. To download it, click here.
Play Movie 1 2 3 DOWNLOAD
Featuring the voice of Gilbert Gottfried
I, Clippit, known to all as "Clippy," am a movie star! (Did you know that most award winners got their starts in Flash animations?) The episodes on this site show the effects of Office XP on a humble paper clip like me. It isn't pretty.
Internet


Microsoft Pregnancy and Child Care
Home
Go To
Find
Options
Help
Microsoft Health Preview
Pregnancy and Child Care
Medical Advisory Board
What's New
Click here for this month's highlights in Microsoft Pregnancy and Child Care.
Library
To browse through illustrated articles on pregnancy, birth, and early child care, click here.
Find By Word
If you know what you're looking for, click here to search the Library by keywords.
Find By Symptom
Click here to find useful information in the Library related to children's symptoms.
Community Center
Have a story to share? Want to send us mail? Click here to access our community bulletin boards.


FIND
By Word
By Symptom
Recent Topics
Questions
Severity of abdominal pain: How severe is the child's abdominal pain?
No
Mild
Moderate
Severe
Don't Know
Start Over
Change
Next >>
Finish
Find By Symptom is finding articles related to the symptom: Abdominal pain. Click Next to continue.
Viral gastroenteritis
Psychosomatic pain
Urinary tract infection
Other

What do Bayesian networks have to offer?

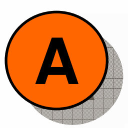
- encoding of the covariation between “input” variables - BN can handle incomplete data sets
- allows one to learn about causal relationships (predictions in the presence of interventions)
- natural way of combining domain knowledge and data as a single model

Three perspectives on dependency modeling:

Model M1:

A and B independent

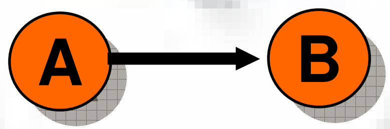
$P(A,B) = P(A)P(B)$



Model M2:

A and B dependent

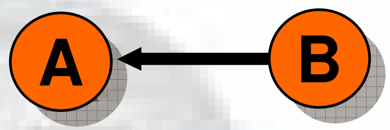
$P(A,B) = P(A)P(B|A)$



Model M3:

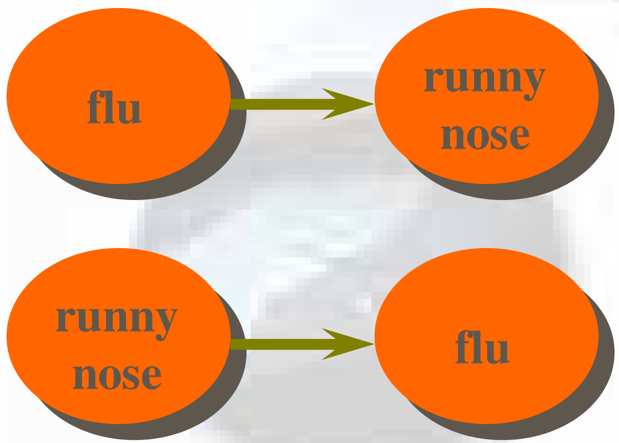
A and B dependent

$P(A,B) = P(B)P(A|B)$



Are the links causal?

- Not necessarily, but causality makes it easier to determine the conditional probabilities.
- Equivalence class=set of BN structures which can used for representing exactly the same set of probability distributions.

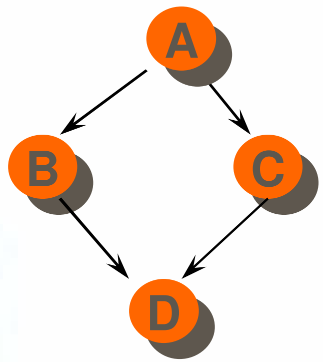


$P(flu, ns) = P(flu)P(rn \mid flu)$

$P(flu, rn) = P(rn)P(flu \mid rn)$

Bayesian networks: basics

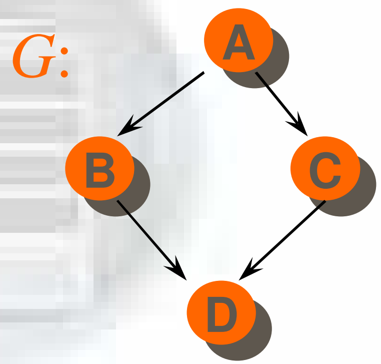
- A Bayesian network is a model of probabilistic dependencies between the domain variables.
- The model can be described as a list of dependencies, but is is usually more convenient to express them in a graphical form as a directed acyclic network.
- The nodes in the network correspond to the domain variables, and the arcs reveal the underlying dependencies, i.e., the hidden structure of the domain of your data.
- The strengths of the dependencies are modeled as conditional probability distributions (not shown in the graph).



Bayesian networks: the textbook definition

- A Bayesian (belief) network representation for a probability distribution *P* on a domain $(X_1,\ldots,X_n)$ is a pair (G,θ), where *G* is a directed acyclic graph whose nodes correspond to the variables $X_1,\ldots,X_n$, and whose topology satisfies the following: each variable X is conditionally independent of all of its non-descendants in *G*, given its set of parents pa_X, and no proper subset of pa_X satisfies this condition. The second component θ is a set consisting of all the conditional probabilities of the form $P(X|pa_X)$.

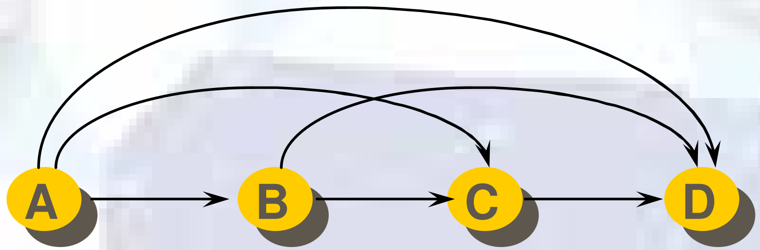
θ = {P(+a), P(+b|+a), P(+b|-a), P(+c|+a), P(+c|-a), P(+d|+b,+c), P(+d|-b,+c), P(+d|+b,-c), P(+d|-b,-c)}



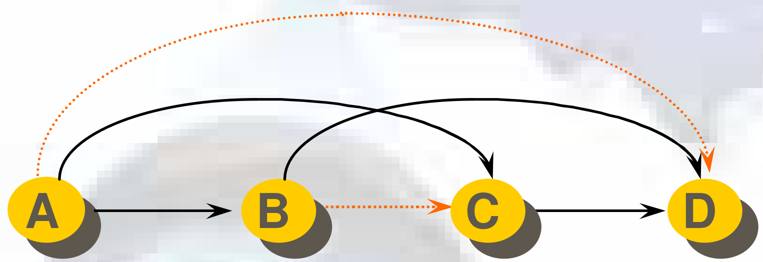
A more intuitive description

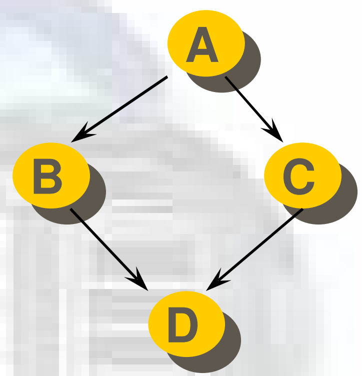
- From Bayes' rule, it follows that

P(A,B,C,D)=P(A)P(B|A)P(C|A,B)P(D|A,B,C)



Assume: P(C|A,B)=P(C|A) and P(D|A,B,C)=P(D|B,C)

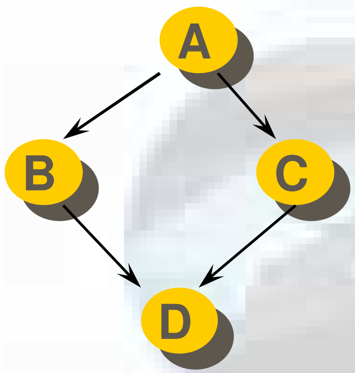




$$P(X_1,\ldots,X_n) = \prod_{i=1}^{n} P(X_i | Pa_i)$$

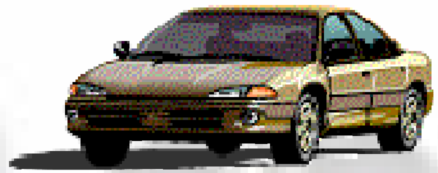
And the point is...?

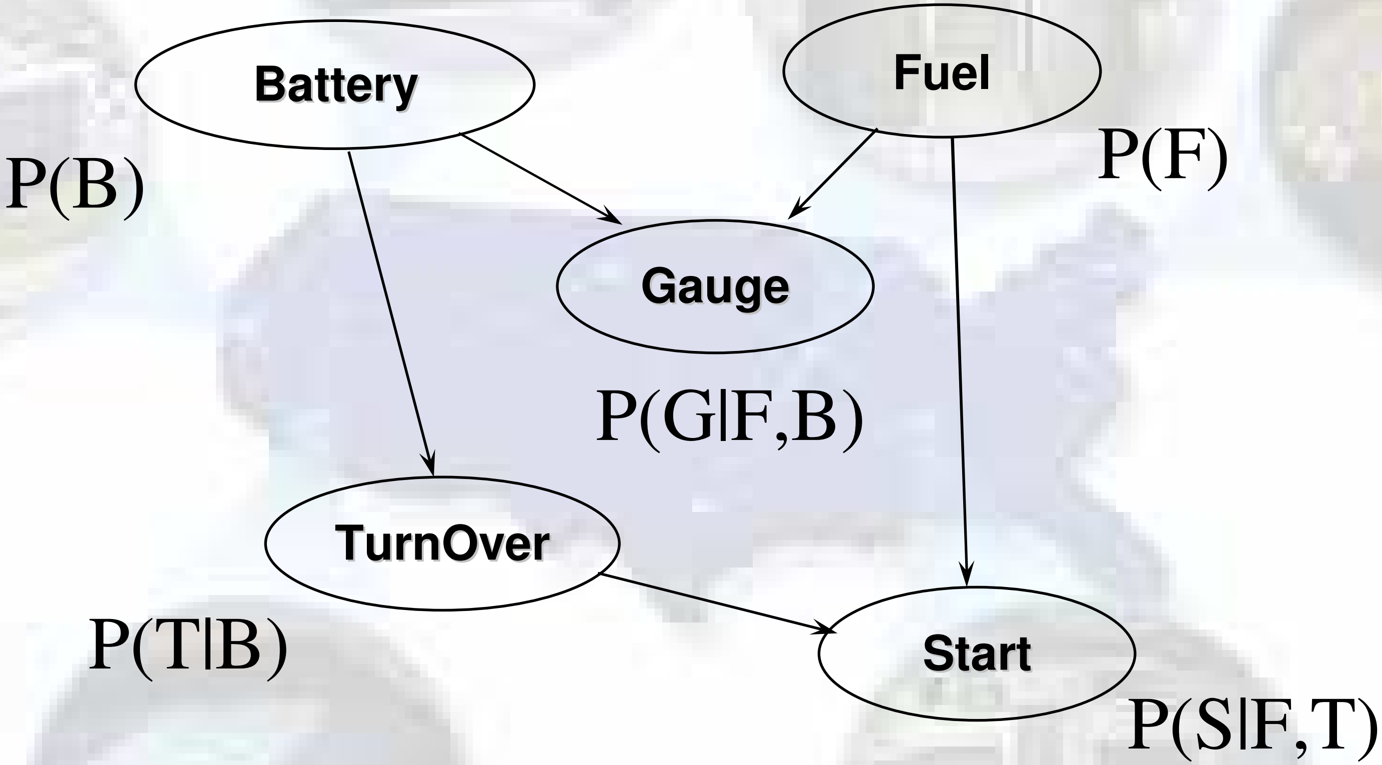
- simple conditional probabilities are easier to determine than the full joint probabilities
- in many domains, the underlying structure corresponds to relatively sparse networks, so only a small number of conditional probabilities is needed



$$P(+a,+b,+c,+d)=P(+a)P(+b|+a)P(+c|+a)P(+d|+b,+c)$$
$$P(-a,+b,+c,+d)=P(-a)P(+b|-a)P(+c|-a)P(+d|+b,+c)$$
$$P(-a,-b,+c,+d)=P(-a)P(-b|-a)P(+c|-a)P(+d|-b,+c)$$
$$P(-a,-b,-c,+d)=P(-a)P(-b|-a)P(-c|-a)P(+d|-b,-c)$$
$$P(-a,-b,-c,-d)=P(-a)P(-b|-a)P(-c|-a)P(-d|-b,-c)$$
$$P(+a,-b,-c,-d)=P(+a)P(-b|+a)P(-c|+a)P(-d|-b,-c)$$
. . .

Bayesian Network





Acyclic directed probabilistic independence network

Bayesian Network

P(T=none) = 0.003
P(T=click)= 0.001
P(T=normal)= 0.996

P(S=yes|T=none) = 0.0
P(S=no|T=none) = 1.0

P(S=yes|T=click) = 0.02
P(S=no|T= click) = 0.98

P(S=yes|T=normal) = 0.97
P(S=no|T=normal) = 0.03

Missing Arcs Encode Conditional Independence

p(T=none) = 0.003
p(T=click)= 0.001
p(T=normal)= 0.996

p(G=not empty) = 0.995
p(G=empty) = 0.005

Defining Bayesian Network Structure

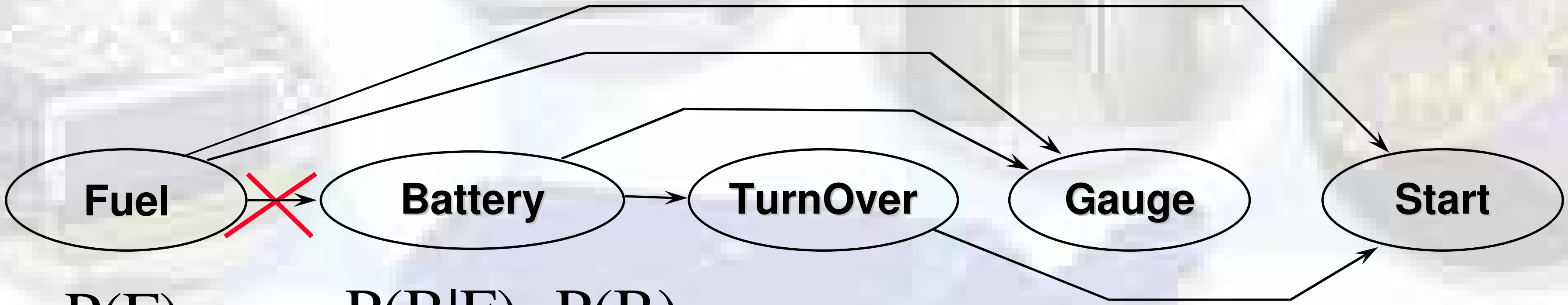
Given an ordering of the variables $(X_1, ..., X_n)$, the parents of X_i, denoted Pa_i, is a subset of $\{X_1,...,X_{i-1}\}$ s.t.

$$P(X_i | X_1, \ldots, X_{i-1}) = P(X_i | Pa_i)$$

$$\Downarrow$$

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | X_1, \ldots, X_{i-1}) = \prod_{i=1}^{n} P(X_i | Pa_i)$$

A Modular Encoding of a Joint Distribution



$P(F)$ $\quad$ $P(B|F)=P(B)$

$P(T|B,F)=P(T|B)$

$P(G|F,B,T)=P(G|F,B)$

$P(S|F,B,T,G)=P(S|F,T)$

$$P(F,B,T,G,S) = P(F)\, P(B|F)\, P(T|B,F)\, P(G|F,B,T)\, P(S|F,B,T,G)$$
$$= P(F)\, P(B)\, P(T|B)\, P(G|F,B)\, P(S|F,T)$$

Local Distributions

$$p(\mathbf{x}|\theta_s) = \prod_{i=1}^{n} p(x_i|\mathbf{pa}_i, \theta_i)$$

parameters (finite #)

local distributions

Examples of Local Distributions

- Local distributions can be e.g.,
  - Multinomial (child and parents are discrete)

$$P(X_i^k \mid Pa_i^j, \theta_i) = \theta_{X_i^k \mid Pa_i^j}$$

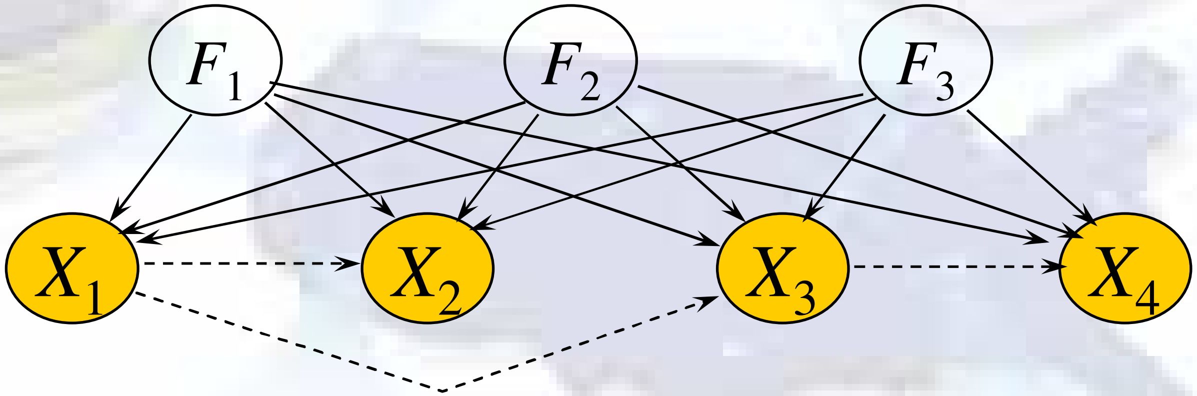
  - Gaussian (child and parents are Gaussian distributions)

$$p(x_i \mid \mathbf{pa}_i, \theta_i) = m_i + \sum_{x_j \in \mathbf{pa}_i} b_{ji} x_j + N(0, \sigma_i^2)$$

  - Mixture (child and parent either Gaussian or multinomial)

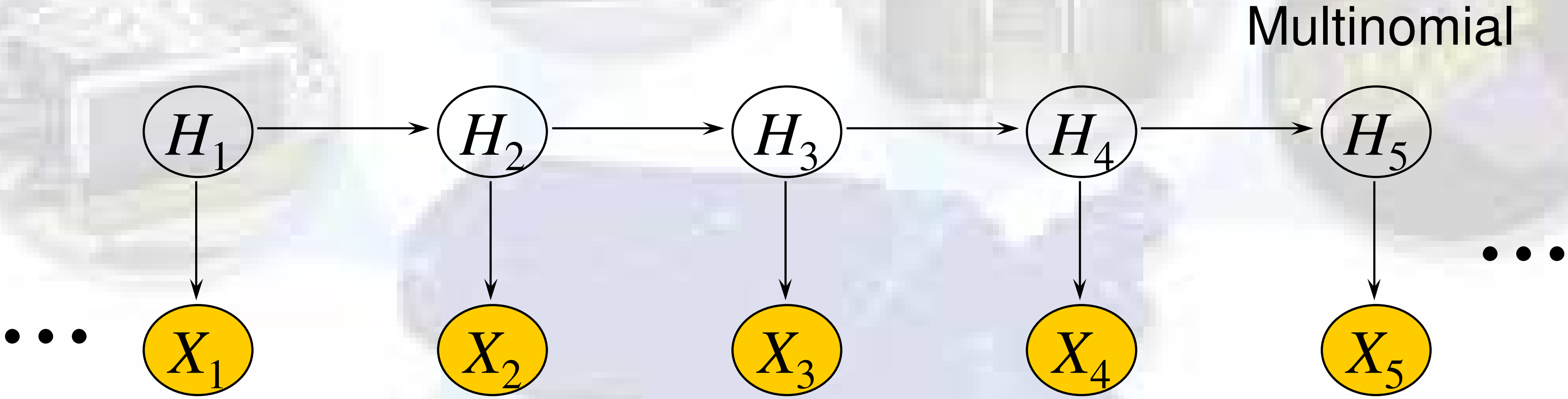
Factor Analysis

Exploratory tool of choice for many applied areas



all Gaussian

Hidden Markov Model



- Multinomial
- Gaussian
- Gaussian mixture

Undirected vs. Directed Models

- Markov network

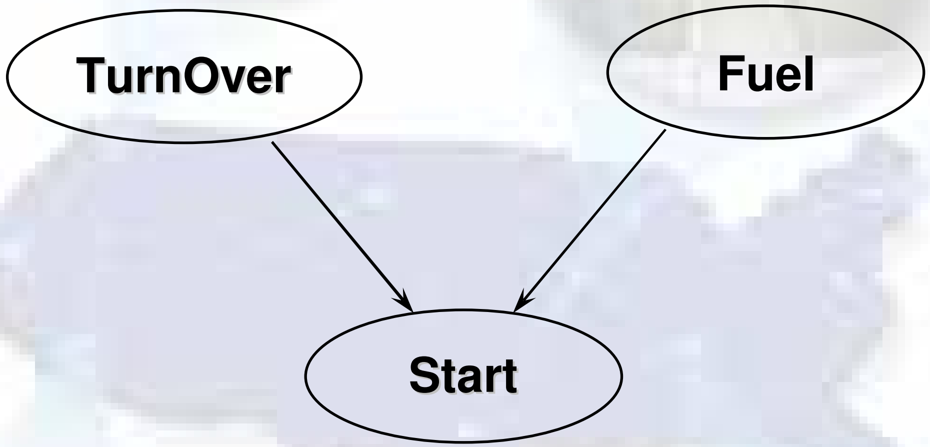
$$X - Y - Z \qquad\qquad X \perp Z \mid Y,\ \ \neg(X \perp Z \mid \emptyset)$$

- Bayesian networks

$$X \rightarrow Y \rightarrow Z \qquad\qquad X \perp Z \mid Y,\ \ \neg(X \perp Z \mid \emptyset)$$

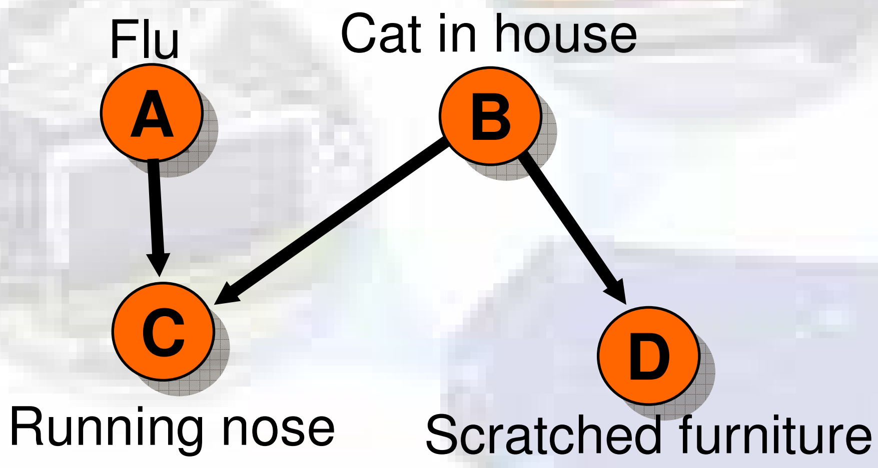
$$X \rightarrow Y \leftarrow Z \qquad\qquad \neg(X \perp Z \mid Y),\ \ X \perp Z$$

Explaining Away (selection bias, Berkson's paradox)



If the car doesn't start, hearing the engine turn over makes no fuel more likely.

Explaining away: an example



P(A=1)=0.05
P(B=1)=0.05
P(C=1|A=0,B=0)=0.001
P(C=1|A=1,B=0)=0.95
P(C=1|A=0,B=1)=0.95
P(C=1|A=1,B=1)=0.99
P(D=1|B=1)=0.99
P(D=1|B=0)=0.1

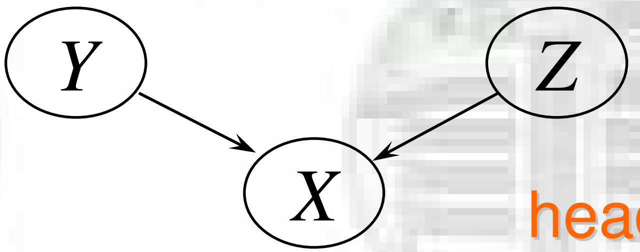
- Given C=1**,** the probability of A=1 is about 51%, and the probability of B=1 is also about 51%
- Given C=1 **and** D=1, the probability of A=1 goes down to 13% while the probability of B=1 goes up to 91%

d-Separation (Pearl 1987)

- Theorem (Verma): X and Y are d-separated by $\mathrm{Z} \;\Rightarrow\; X \perp Y \mid \mathrm{Z}$.
- Theorem (Geiger and Pearl): If X and Y are not d-separated by Z, then there exists an assignment of the probabilities to the BN such that $\neg\,(\,X \perp Y \mid \mathrm{Z}\,)$.

d-Separation

- A *trail* in a BN is a sequence of edges in the corresponding undirected graph that forms a cycle-free path
- A node x is a head-to-head node along a trail if there are two consecutive arcs Y **->** X and X **<-** Z on that trail



head-to-head node

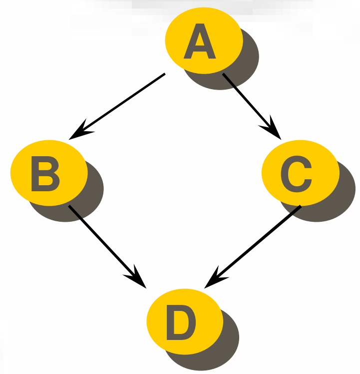
d-Separation

- Nodes X and Y are d-connected by nodes Z along a trail from X to Y if
  - every head-to-head node along the trail is in Z or has a descendant in Z
  - every other node along the trail is not in Z

> Nodes ***X*** and ***Y*** are d-separated by nodes Z if they are not d-connected by Z along any trail from ***X*** to ***Y***

Reading out the dependencies

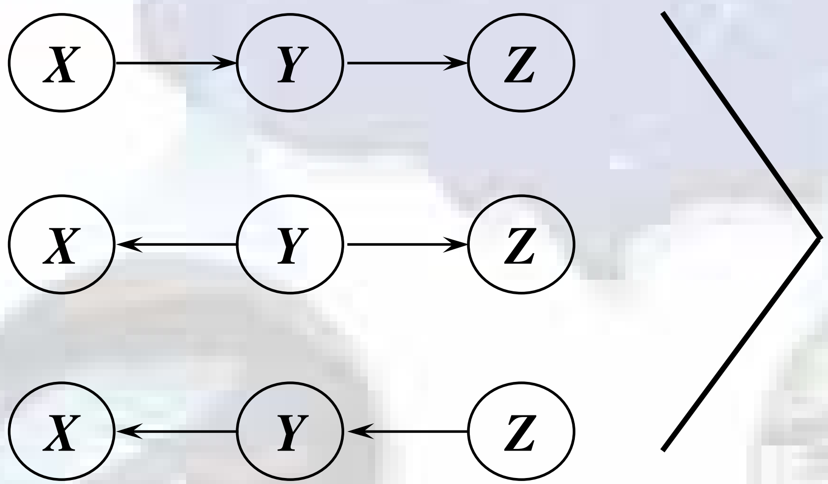
- The Bayesian network on the right represents the following list of dependencies:
  - A and B are dependent on each other no matter what we know and what we don't know about C or D (or both).
  - A and C are dependent on each other no matter what we know and what we don't know about B or D (or both).
  - B and D are dependent on each other no matter what we know and what we don't know about A or C (or both).
  - C and D are dependent on each other no matter what we know and what we don't know about A or B (or both).
  - A and D are dependent on each other if we do not know both B and C.
  - B and C are dependent on each other if we know D or if we do not know D and also do not know A.



Equivalent Network Structures

Two network structures for domain X are independence equivalent if they encode the same set of conditional independence statements

Example:

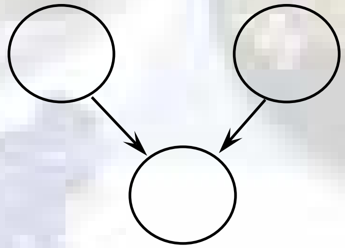


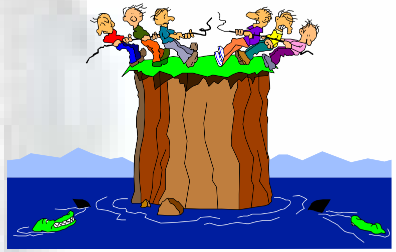
$$X \perp Z \mid Y$$

Equivalent Network Structures

Verma (1990): Two network structures for U are independence equivalent if and only if

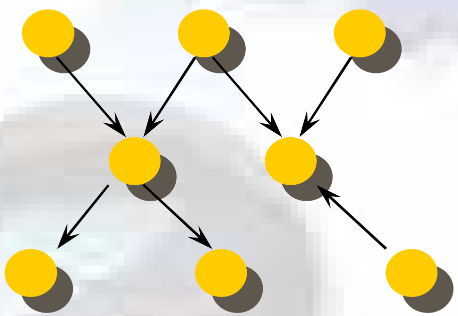
- They have the same skeleton
- They have the same v-structures



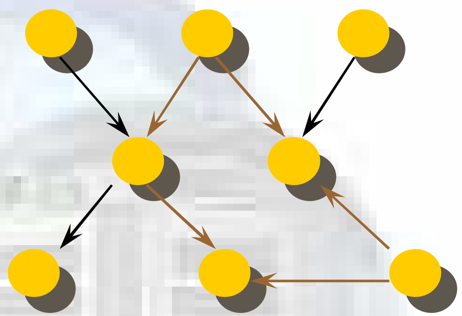


Singly-connected BNs

- a singly connected BN = polytree (disregarding the arc directions, no two nodes can be connected with more than one path).

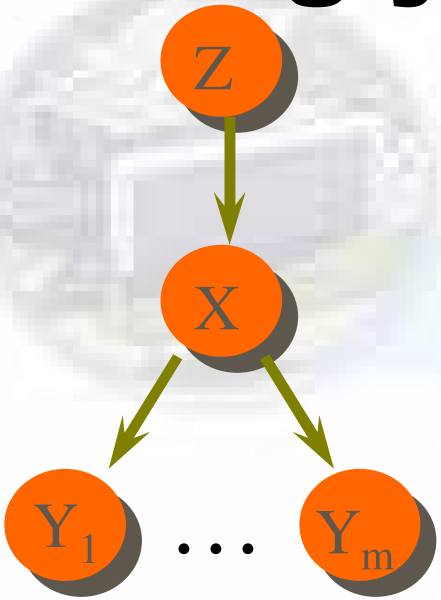


singly-connected



multi-connected

Probabilistic reasoning in singly-connected BNs



$$P(X = x|E) \propto P(E_{X-}|X = x)P(X = x|E_{X+})$$

$$P(E_{X-}|X = x) = \prod_{Y} P(E_{Y-}|X = x)$$

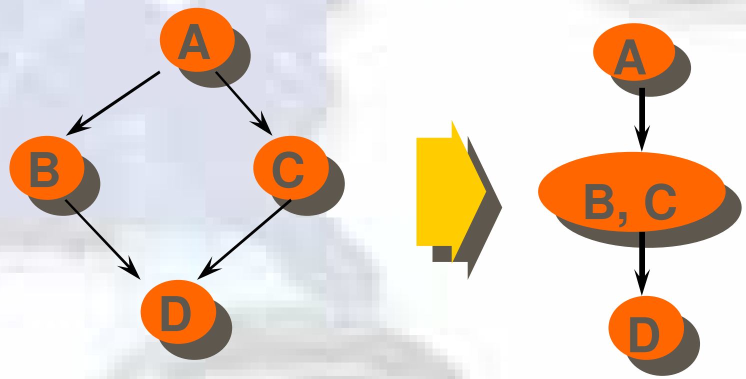
$$P(E_{Y-}|X = x) = \sum_{y} P(Y = y|X = x)P(E_{Y-}|Y = y)$$

$$P(X = x|E_{X+}) = \sum_{z} P(X = x|Z = z)P(Z = z|E_{Z+})$$

- a computationally efficient message-passing scheme: time requirement linear in the number of conditional probabilities in θ.

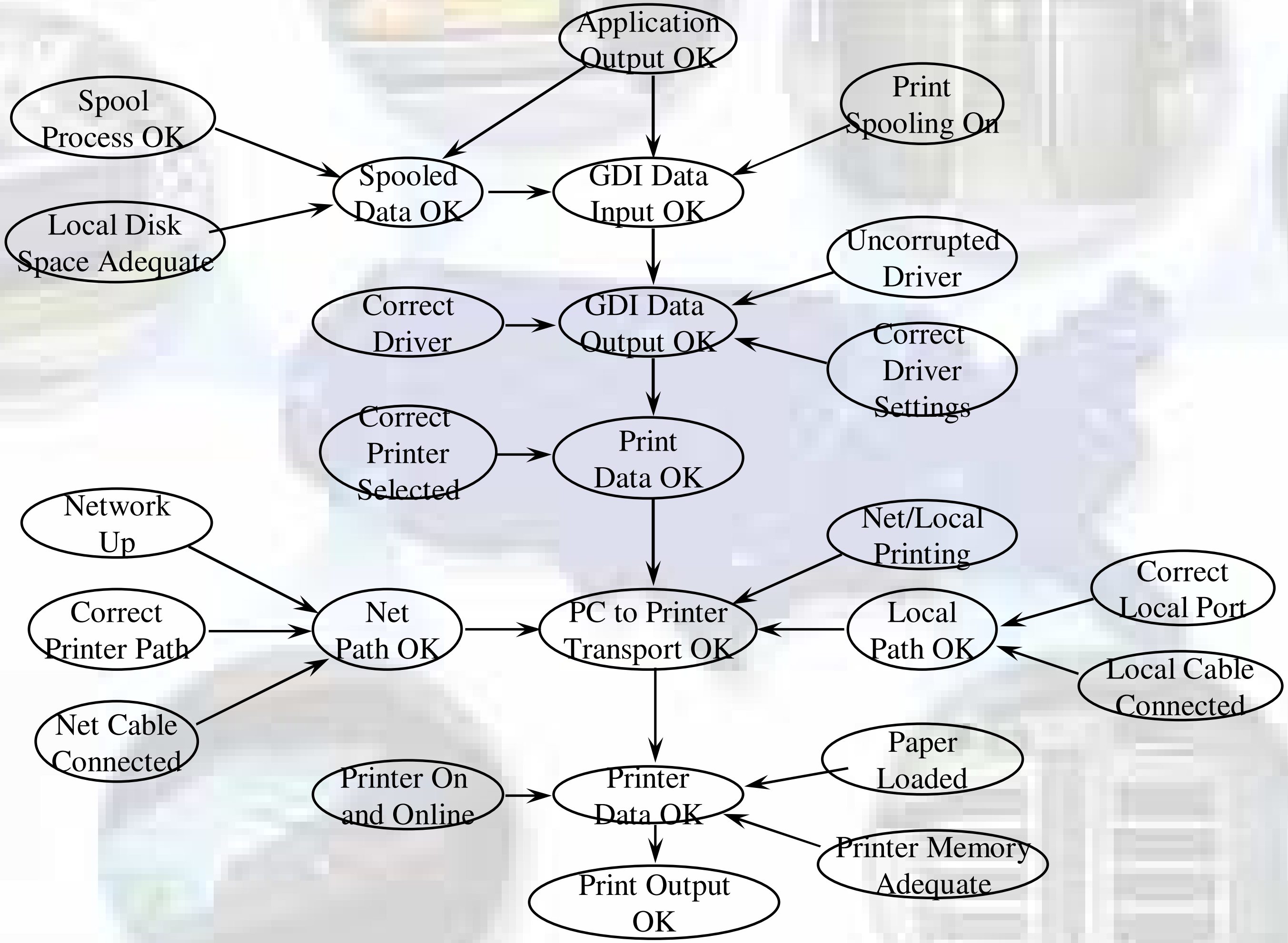
Probabilistic reasoning in multi-connected BNs

- generally not computationally feasible as the problem has been shown to be NP-hard (Cooper 1990, Shimony 1994).
- exact methods:
  - clustering
  - conditioning
  - algebraic methods



- approximative methods:
  - stochastic sampling algorithms
  - deterministic approximations with bounded accuracy

Print Troubleshooter (W '95)



So let us play....