Lecture 27 of 42

Reasoning under Uncertainty: Introduction to Graphical Models, Part 1 of 2

William H. Hsu
Department of Computing and Information Sciences, KSU

KSOL course page: http://snipurl.com/v9v3
Course web site: http://www.kddresearch.org/Courses/CIS730
Instructor home page: http://www.cis.ksu.edu/~bhsu

Reading for Next Class:

Sections 14.3 – 14.5, p. 500 - 518, Russell & Norvig 2nd edition

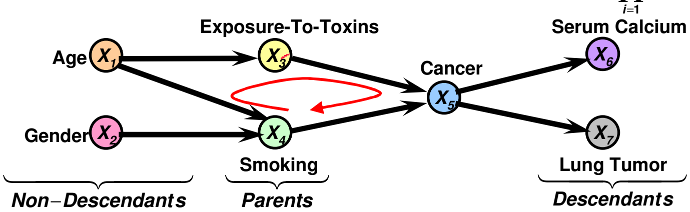
Lecture Outline

- Reading for Next Class: Sections 14.3 – 14.5 (p. 500 – 518), R&N 2e
- Last Class: Uncertainty, Chapter 13 (p. 462 - 489)
- Today: Graphical Models, 14.1 – 14.2 (p. 492 – 499), R&N 2e
- Coming Week: More Applied Probability, Graphical Models

Graphical Models of Probability

- Conditional Independence
  - *X* is conditionally independent (CI) from *Y* given *Z* iff $P(X \mid Y, Z) = P(X \mid Z)$ for all values of *X*, *Y*, and *Z*
  - Example: *P*(*Thunder* | *Rain*, *Lightning*) = *P*(*Thunder* | *Lightning*) ⇔ $T \perp R \mid L$
- Bayesian (Belief) Network
  - Acyclic directed graph model $B = (V, E, \Theta)$ representing *CI assertions* over Θ
  - Vertices (nodes) *V*: denote events (each a random variable)
  - Edges (arcs, links) *E*: denote conditional dependencies
- Markov Condition for BBNs (Chain Rule):
- Example BBN

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid parents(X_i))$$



P(*20s*, *Female*, *Low*, *Non-Smoker*, *No-Cancer*, *Negative*, *Negative*)
= *P*(*T*) · *P*(*F*) · *P*(*L* | *T*) · *P*(*N* | *T*, *F*) · *P*(*N* | *L*, *N*) · *P*(*N* | *N*) · *P*(*N* | *N*)

Semantics of Bayesian Networks

"Global" semantics defines the full joint distribution as the product of the local conditional distributions:

$$\mathbf{P}(X_1, \ldots, X_n) = \Pi_{i=1}^{n} \mathbf{P}(X_i \mid Parents(X_i))$$

e.g., $P(J \wedge M \wedge A \wedge \neg B \wedge \neg E)$ is given by??
$= P(\neg B) P(\neg E) P(A \mid \neg B \wedge \neg E) P(J \mid A) P(M \mid A)$

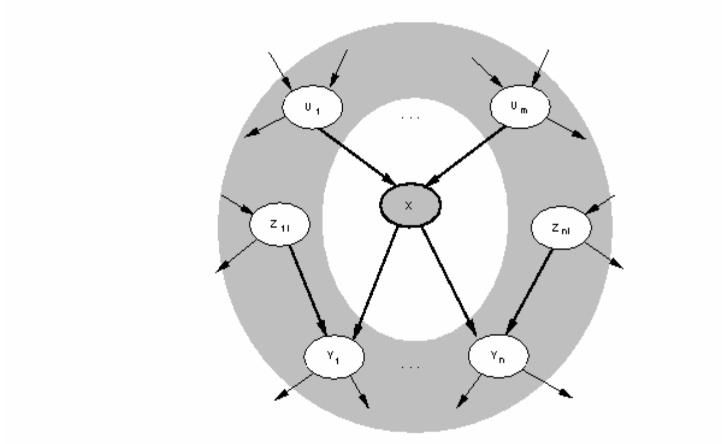
"Local" semantics: each node is conditionally independent of its nondescendants given its parents

Theorem: Local semantics ⇔ global semantics

CIS 530 / 730
Artificial Intelligence
Lecture 27 of 42
Computing & Information Sciences
Kansas State University

Markov Blanket

Each node is conditionally independent of all others given its
<u>Markov blanket</u>: parents + children + children's parents



Constructing Bayesian Networks: Chain Rule of Inference

Need a method such that a series of locally testable assertions of conditional independence guarantees the required global semantics

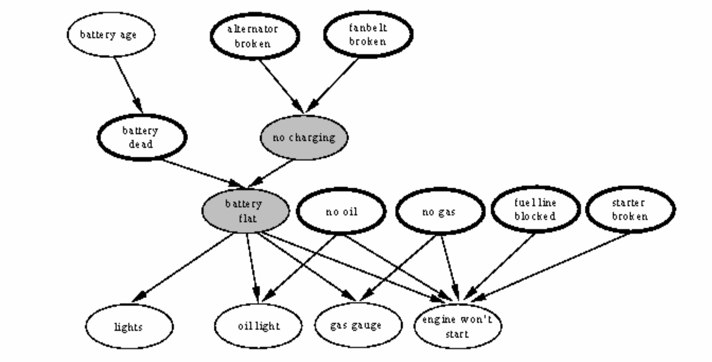
1. Choose an ordering of variables $X_1, \ldots, X_n$
2. For $i = 1$ to n
   add X_i to the network
   select parents from $X_1, \ldots, X_{i-1}$ such that
   $\mathbf{P}(X_i | Parents(X_i)) = \mathbf{P}(X_i | X_1, \ldots, X_{i-1})$

This choice of parents guarantees the global semantics:

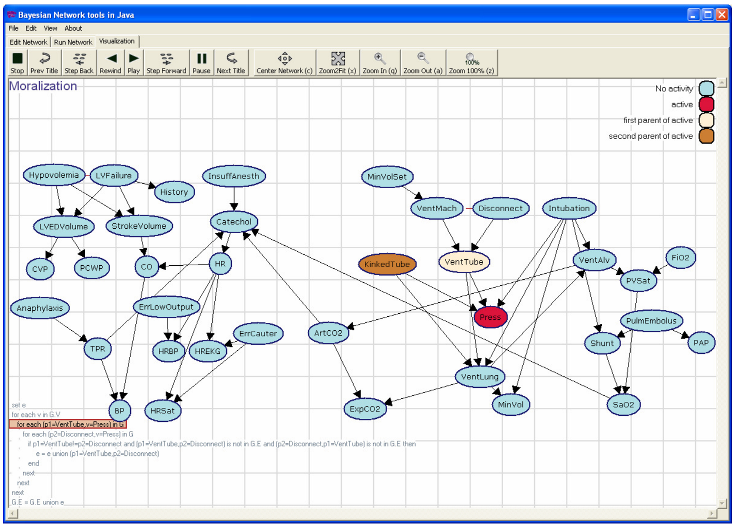
$\mathbf{P}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \mathbf{P}(X_i | X_1, \ldots, X_{i-1})$ (chain rule)
$= \prod_{i=1}^{n} \mathbf{P}(X_i | Parents(X_i))$ by construction

Evidential Reasoning: Example – Car Diagnosis

Initial evidence: engine won't start
Testable variables (thin ovals), diagnosis variables (thick ovals)
Hidden variables (shaded) ensure sparse structure, reduce parameters

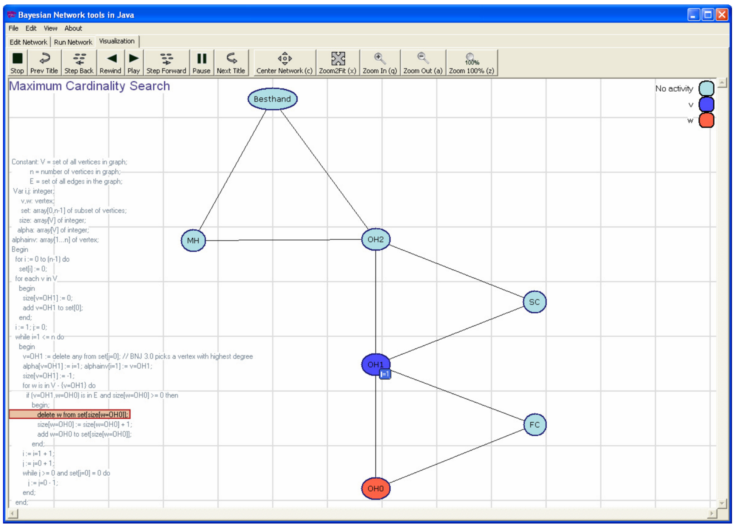


BNJ Visualization [1]
Pseudo-Code Annotation (Code Page)



ALARM Network

BNJ Visualization [2] Network



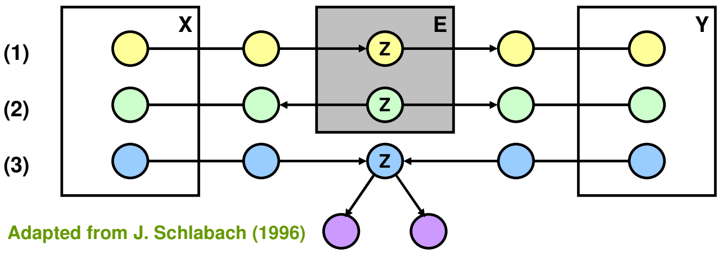
Poker Network

Graphical Models Overview [2]: Markov Blankets & *d*-Separation

Motivation: The conditional independence status of nodes within a BBN might change as the availability of evidence *E* changes. *Direction-dependent separation* (*d-separation*) is a technique used to determine conditional independence of nodes as evidence changes.

Definition: A set of evidence 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* if one of three conditions holds:



Adapted from J. Schlabach (1996)

Graphical Models Overview [3]: Inference Problem

Typically, we are interested in
the posterior joint distribution of the query variables $\mathbf{Y}$
given specific values $\mathbf{e}$ for the evidence variables $\mathbf{E}$

Let the hidden variables be $\mathbf{H} = \mathbf{X} - \mathbf{Y} - \mathbf{E}$

Then the required summation of joint entries is done by summing out the hidden variables:

$$\mathbf{P}(\mathbf{Y}|\mathbf{E}=\mathbf{e}) = \alpha\mathbf{P}(\mathbf{Y},\mathbf{E}=\mathbf{e}) = \alpha\Sigma_{\mathbf{h}}\mathbf{P}(\mathbf{Y},\mathbf{E}=\mathbf{e},\mathbf{H}=\mathbf{h})$$

The terms in the summation are joint entries because $\mathbf{Y}$, $\mathbf{E}$, and $\mathbf{H}$ together exhaust the set of random variables

Obvious problems:
1) Worst-case time complexity $O(d^n)$ where d is the largest arity
2) Space complexity $O(d^n)$ to store the joint distribution
3) How to find the numbers for $O(d^n)$ entries???

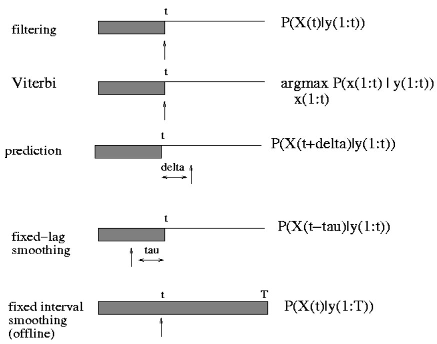
Multiply-connected case: exact, approximate inference are #P-complete

Adapted from slide © 2004 S. Russell & P. Norvig. Reused with permission.

Other Topics in Graphical Models [1]: Temporal Probabilistic Reasoning

Adapted from Murphy (2001), Guo (2002)

- **Goal: Estimate** $P(X_t^i \mid y_{1 \ldots r})$
- **Filtering:** $r = t$
  - **Intuition: infer current state from observations**
  - **Applications: signal identification**
  - **Variation: Viterbi algorithm**
- **Prediction:** $r < t$
  - **Intuition: infer future state**
  - **Applications: prognostics**
- **Smoothing:** $r > t$
  - **Intuition: infer past hidden state**
  - **Applications: signal enhancement**
- **CF Tasks**
  - **Plan recognition by smoothing**
  - **Prediction cf. *WebCANVAS* – Cadez *et al.* (2000)**



Other Topics in Graphical Models [2]: Learning Structure from Data

- **General-Case BN Structure Learning: *Use Inference to Compute Scores***
- **Optimal Strategy: Bayesian Model Averaging**
  - **Assumption: models $h \in H$ are mutually exclusive and exhaustive**
  - ***Combine predictions of models in proportion to marginal likelihood***
    - **Compute conditional probability of hypothesis h given observed data D**
    - ***i.e.,* compute expectation over unknown h for unseen cases**
    - **Let $h \equiv$ structure, parameters $\Theta \equiv$ CPTs**

$$P\left(\vec{x}^{(m+1)} \mid D\right) = P\left(x_1, x_2, \ldots, x_n \mid \vec{x}^{(1)}, \vec{x}^{(2)}, \ldots, \vec{x}^{(m)}\right)$$

$$= \sum_{h \in H} P\left(\vec{x}^{(m+1)} \mid D, h\right) \cdot P(h \mid D)$$

Posterior Score

Marginal Likelihood

Prior over Parameters

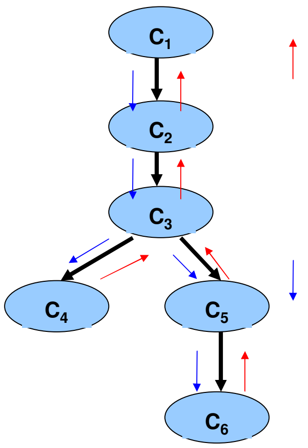
$$P(h \mid D) \propto P(D \mid h) \cdot P(h)$$

$$= P(h) \cdot \int P(D \mid h, \Theta) \cdot P(\Theta \mid h)\, d\Theta$$

Prior over Structures

Likelihood

Propagation Algorithm in Singly-Connected BNs – Pearl (1983)



Upward (child-to-parent) λ messages

$\Psi'(C_i')$ modified during λ message-passing phase

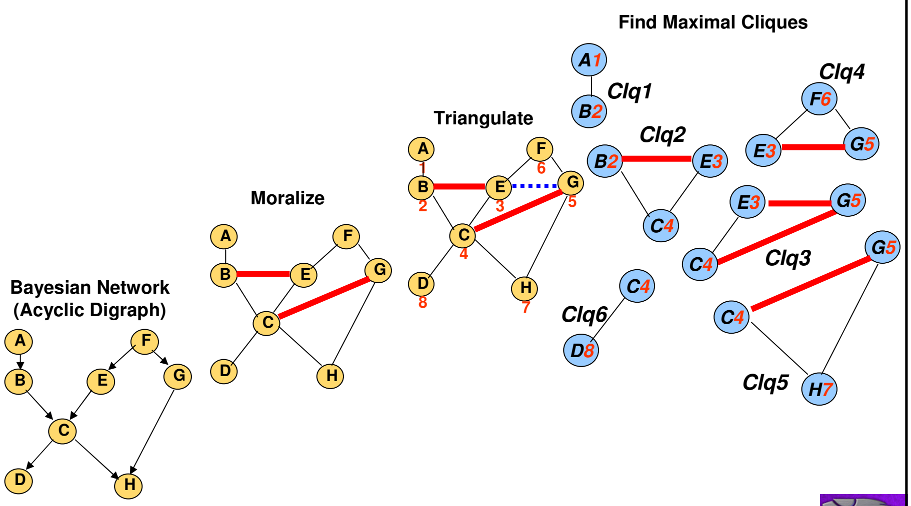
Downward π messages

$P'(C_i')$ is computed during π message-passing phase

Multiply-connected case: exact, approximate inference are #P-complete
(counting problem is #P-complete iff decision problem is NP-complete)

Adapted from Neapolitan (1990), Guo (2000)

Inference by Clustering [1]: Moralization, Triangulation, Cliques



Adapted from Neapolitan (1990), Guo (2000)

Inference by Clustering [2]: Junction Tree Algorithm

Input: list of cliques of **triangulated, moralized graph G_u**

Output:

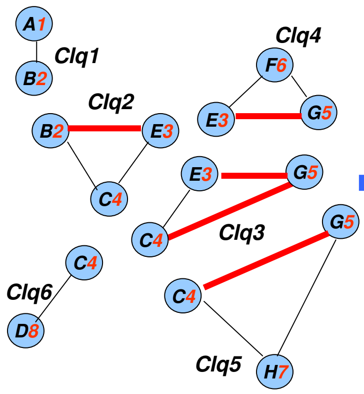
- **Tree of cliques**
- **Separators nodes S_i**,
- Residual nodes R_i and **potential probability $\Psi(Clq_i)$** for all cliques

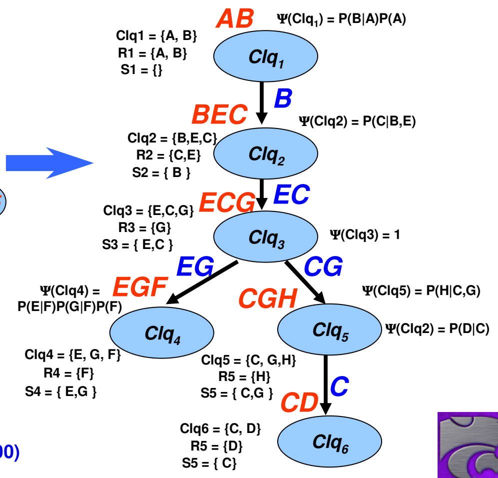
Algorithm:

1. $S_i = Clq_i \cap (Clq_1 \cup Clq_2 \cup \ldots \cup Clq_{i-1})$
2. $R_i = Clq_i - S_i$
3. If $i > 1$ then identify a $j < i$ such that Clq_j is a parent of Clq_i
4. Assign each node v to a unique clique Clq_i that $v \cup c(v) \subseteq Clq_i$
5. Compute $\Psi(Clq_i) = \Pi_{f(v)\ Clqi} = P(v \mid c(v))$ {1 if no v is assigned to Clq_i}
6. Store Clq_i, R_i, S_i, and $\Psi(Clq_i)$ at each vertex in the tree of cliques

Inference by Clustering [2]: Clique Tree Operations

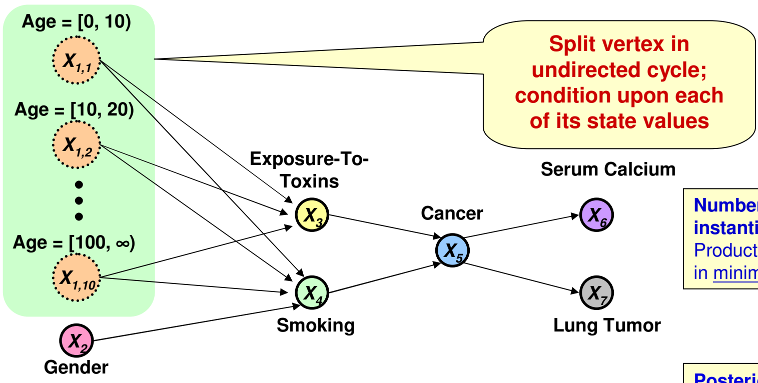
R_i: residual nodes
S_i: separator nodes
$\Psi(Clq_i)$: potential probability of Clique i





Adapted from Neapolitan (1990), Guo (2000)

Inference by Loop Cutset Conditioning



Number of network instantiations: Product of arity of nodes in minimal loop cutset

Posterior: marginal conditioned upon cutset variable values

- **Deciding Optimal Cutset: NP-*hard***
- **Current Open Problems**
  - **Bounded cutset conditioning: ordering heuristics**
  - **Finding randomized algorithms for loop cutset optimization**

Inference by Variable Elimination [1]: Factoring Operations

Enumeration is inefficient: repeated computation
e.g., computes $P(J = true|a)P(M = true|a)$ for each value of e

Variable elimination: carry out summations right-to-left,
storing intermediate results (factors) to avoid recomputation

$$\begin{aligned}
&\mathbf{P}(B|J = true, M = true)\\
&\quad = \alpha \underbrace{\mathbf{P}(B)}_{B} \Sigma_e \underbrace{P(e)}_{E} \Sigma_a \underbrace{\mathbf{P}(a|B,e)}_{A} \underbrace{P(J = true|a)}_{J} \underbrace{P(M = true|a)}_{M}\\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a \mathbf{P}(a|B,e) P(J = true|a) f_M(a)\\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a \mathbf{P}(a|B,e) f_J(a) f_M(a)\\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a f_A(a,b,e) f_J(a) f_M(a)\\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) f_{\bar{A}JM}(b,e) \text{ (sum out } A)\\
&\quad = \alpha \mathbf{P}(B) f_{\bar{E}\bar{A}JM}(b) \text{ (sum out } E)\\
&\quad = \alpha f_B(b) \times f_{\bar{E}\bar{A}JM}(b)
\end{aligned}$$

Adapted from slide © 2004 S. Russell & P. Norvig. Reused with permission.

Inference by Variable Elimination [2]: Factoring Operations

Pointwise product of factors f_1 and f_2:

$$\begin{aligned}
&f_1(x_1, \ldots, x_j, y_1, \ldots, y_k) \times f_2(y_1, \ldots, y_k, z_1, \ldots, z_l)\\
&\quad = f(x_1, \ldots, x_j, y_1, \ldots, y_k, z_1, \ldots, z_l)
\end{aligned}$$

E.g., $f_1(a,b) \times f_2(b,c) = f(a,b,c)$

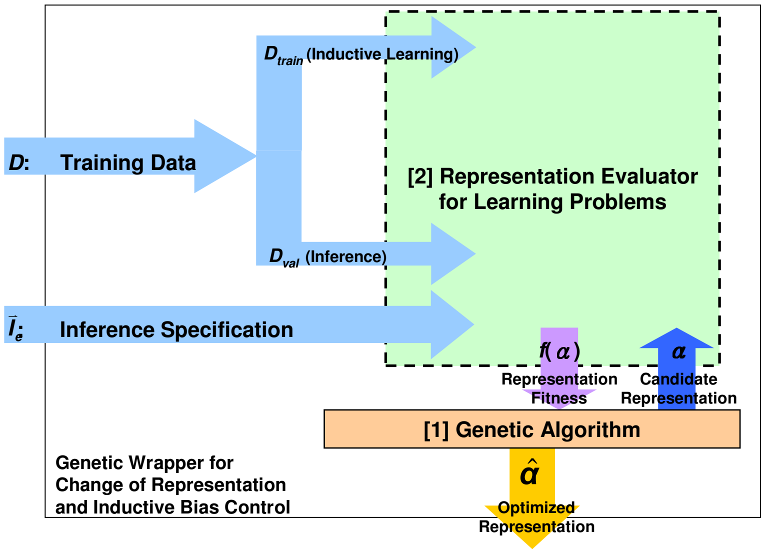
Summing out a variable from a product of factors: move any constant factors outside the summation:

$$\Sigma_x f_1 \times \cdots \times f_k = f_1 \times \cdots \times f_i \, \Sigma_x \, f_{i+1} \times \cdots \times f_k = f_1 \times \cdots \times f_i \times f_{\bar{X}}$$

assuming $f_1, \ldots, f_i$ do not depend on X

Adapted from slide © 2004 S. Russell & P. Norvig. Reused with permission.

Genetic Algorithms for Parameter Tuning in Learning



References: Graphical Models & Inference

- **Graphical Models**
  - Bayesian (Belief) Networks tutorial – Murphy (2001)
    http://www.cs.berkeley.edu/~murphyk/Bayes/bayes.html
  - Learning Bayesian Networks – Heckerman (1996, 1999)
    http://research.microsoft.com/~heckerman
- **Inference Algorithms**
  - Junction Tree (Join Tree, L-S, *Hugin*): Lauritzen & Spiegelhalter (1988)
    http://citeseer.nj.nec.com/huang94inference.html
  - (Bounded) Loop Cutset Conditioning: Horvitz & Cooper (1989)
    http://citeseer.nj.nec.com/shachter94global.html
  - Variable Elimination (Bucket Elimination, *ElimBel*): Dechter (1986)
    http://citeseer.nj.nec.com/dechter96bucket.html
  - Recommended Books
    - Neapolitan (1990) – *out of print*; see Pearl (1988), Jensen (2001)
    - Castillo, Gutierrez, Hadi (1997)
    - Cowell, Dawid, Lauritzen, Spiegelhalter (1999)
  - Stochastic Approximation
    http://citeseer.nj.nec.com/cheng00aisbn.html

Terminology

- **Uncertain Reasoning**
  - Ability to perform inference in presence of uncertainty about
    - premises
    - rules
  - Nondeterminism
- **Representations for Uncertain Reasoning**
  - Probability: measure of belief in sentences
    - Founded on Kolmogorov axioms
    - prior, joint *vs.* conditional
    - Bayes's theorem: P(A | B) = (P(B | A) * P(A)) / P(B)
  - Graphical models: graph theory + probability
  - Dempster-Shafer theory: upper and lower probabilities, reserved belief
  - Fuzzy representation (sets), fuzzy logic: degree of membership
  - Others
    - Truth maintenance system: logic-based network representation
    - Endorsements: evidential reasoning mechanism

Summary Points

- **Last Class: Reasoning under Uncertainty and Probability**
  - Uncertainty is pervasive
    - Planning
    - Reasoning
    - Learning (later)
  - What are we uncertain about?
    - Sensor error
    - Incomplete or faulty domain theory
    - "Nondeterministic" environment
- **Today: Graphical Models**
- **Coming Week: More Applied Probability**
  - Graphical models as KR for uncertainty: Bayesian networks, *etc.*
  - Some inference algorithms for Bayes nets

CIS 530 / 730
Artificial Intelligence
Lecture 27 of 42
Computing & Information Sciences
Kansas State University