Koller & Friedman Chapter 13

Structure Learning:
the good, the bad, the ugly

Graphical Model – 10708
Carlos Guestrin
Carnegie Mellon University

October 24th, 2005

Announcements

- Project feedback by e-mail soon

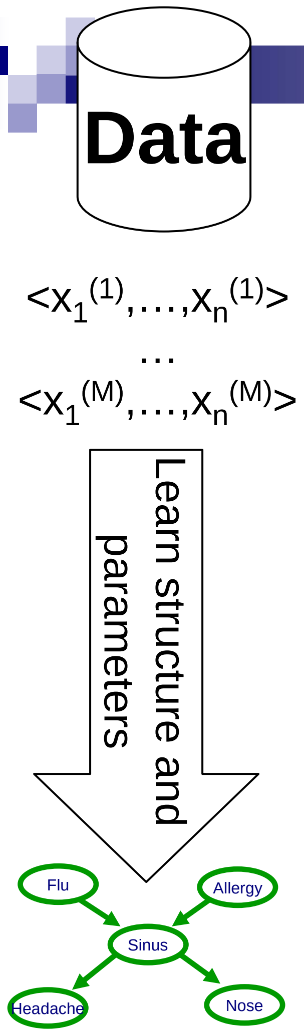
Where are we?

- Bayesian networks
- Undirected models
- Exact inference in GMs
  - Very fast for problems with low tree-width
  - Can also exploit CSI and determinism

Parameter Learning

- Learning GMs
  - Given structure, estimate parameters
    - Maximum likelihood estimation (just counts for BNs)
    - Bayesian learning
    - MAP for Bayesian learning
  - What about learning structure?

Learning the structure of a BN



- **Constraint-based approach**
  - BN encodes conditional independencies
  - Test conditional independencies in data
  - Find an I-map
- **Score-based approach**
  - Finding a structure and parameters is a density estimation task
  - Evaluate model as we evaluated parameters
    - Maximum likelihood
    - Bayesian
    - etc.

Remember: Obtaining a P-map? September 21st lecture… ☺

- Given the independence assertions that are true for *P*
  - Obtain skeleton
  - Obtain immoralities
- From skeleton and immoralities, obtain every (and any) BN structure from the equivalence class

Ask indep. queries : $(X \perp Y \mid U)$?

- **Constraint-based approach**:
  - Use Learn PDAG algorithm
  - Key question: **Independence test**

Independence tests

- Statistically difficult task!
- Intuitive approach: **Mutual information**

$$I(X_i, X_j) = \sum_{x_i, x_j} P(x_i, x_j) \log \frac{P(x_i, x_j)}{P(x_i)P(x_j)}$$

- Mutual information and independence:
  - X_i and X_j independent if and only if $I(X_i, X_j) = 0$

$(X \perp Y \mid U)$?

- Conditional mutual information:

$P(X, Y \mid U) = P(X \mid U) \cdot P(Y \mid U)$: $I(X, Y \mid U) = \sum_{x,y,u} P(x,y,u) \cdot \log \frac{P(x,y \mid u)}{P(x \mid u) \cdot P(y \mid u)}$

$\forall X, Y, U$

Independence tests and the constraint based approach

- Using the data *D*
  - Empirical distribution: $\hat{P}(x_i, x_j) = \frac{\text{Count}(x_i, x_j)}{M}$
  - Mutual information: $\hat{I}(X_i, X_j) = \sum_{x_i, x_j} \hat{P}(x_i, x_j) \log \frac{\hat{P}(x_i, x_j)}{\hat{P}(x_i)\hat{P}(x_j)}$
  - Similarly for conditional MI $\hat{I}(X_i, X_j | U)$
- Use learning PDAG algorithm:
  - When algorithm asks: (X⊥Y|**U**)?

    $\hat{I}(X, Y | U) < t$ indep.

    $\hat{I}(X, Y | U) \geq t$ not indep.
- Must check if statistically-significficant
  - Choosing *t*
  - See reading…

Score-based approach

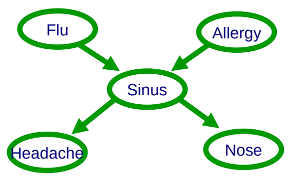
Data

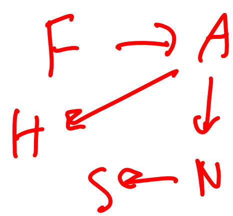
$<x_1^{(1)},\dots,x_n^{(1)}>$

…

$<x_1^{(M)},\dots,x_n^{(M)}>$

Possible structures





Learn parameters

M.L.

Bayesian

Score structure

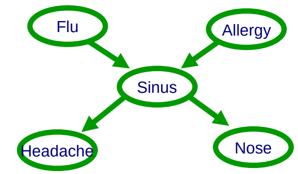
10^{-5}

10^{-12}

pick largest

Information-theoretic interpretation of maximum likelihood

M - datapoint



- Given structure, log likelihood of data:

$$\log P(D \mid \theta_G, G) = \log \prod_{x^i} P(x^{(i)} \mid \theta_G, G)$$

$$= \sum_{x^{(i)}} \log P(x^{(i)} \mid \theta_G, G)$$

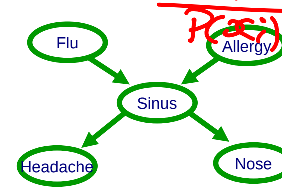
$$= \sum_{x^{(i)}} \sum_{j} \log P(x_j^{(i)} \mid Pa_{x_j}^{(i)}, \theta_{x_j \mid Pa_{x_j}, G}, G)$$

$$= M \sum_{j} \sum_{x_j, Pa_{x_j}} \frac{\text{Count}(x_j, Pa_{x_j})}{M} \log P(x_j \mid Pa_{x_j}, \theta, G)$$

$$= M \sum_{j} \sum_{x_j, Pa_{x_j}} \hat{P}(x_j, Pa_{x_j}) \log P(x_j \mid Pa_{x_j}, \theta, G)$$

Information-theoretic interpretation of maximum likelihood 2

$$\hat{I}(X_i, X_j) = \sum_{x_i, x_j} \hat{P}(x_i, x_j) \log \frac{\hat{P}(x_i, x_j)}{\hat{P}(x_i)\hat{P}(x_j)}$$



- Given structure, log likelihood of data:

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_{i=1}^{n} \sum_{x_i, \mathbf{Pa}_{x_i,\mathcal{G}}} \hat{P}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) \log \hat{P}(x_i \mid \mathbf{Pa}_{x_i,\mathcal{G}})$$

$$= M \sum_i \sum_{x_i, Pa_{x_i}} \hat{P}(x_i, Pa_{x_i}) \log \frac{\hat{P}(x_i, Pa_{x_i})}{\hat{P}(Pa_{x_i})} \frac{P(x_i)}{P(x_i)},$$

$$= M \sum_i \hat{I}(X_i, Pa_{X_i}) - M \sum_i \hat{H}(X_i)$$

$$\hat{I}(X_i, Pa_{X_i})$$

$$\hat{H}(X_i) = -\sum_{x_i} \hat{P}(x_i) \log \hat{P}(x_i)$$

Decomposable score

- Log data likelihood

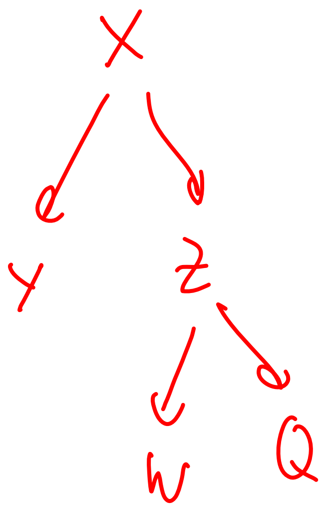
$$\log \widehat{P}(\mathcal{D} \mid \theta_{\mathcal{G}}, \mathcal{G}) = M \sum_i \widehat{I}(\cancel{x}_i, \mathbf{Pa}_{\cancel{x}_i,\mathcal{G}}) - M \sum_i \widehat{H}(X_i)$$

- Decomposable score: node & its parents
  - Decomposes over families in BN (node and its parents)
  - Will lead to significant computational efficiency!!!
  - Score($G : D$) = $\sum_i$ FamScore(X_i|$\mathbf{Pa}_{X_i}$: D)

$$= \sum_i M\left[I(X_i, Pa_{X_i}) - H(X_i)\right]$$

How many trees are there?

Nonetheless – Efficient optimal algorithm finds best tree



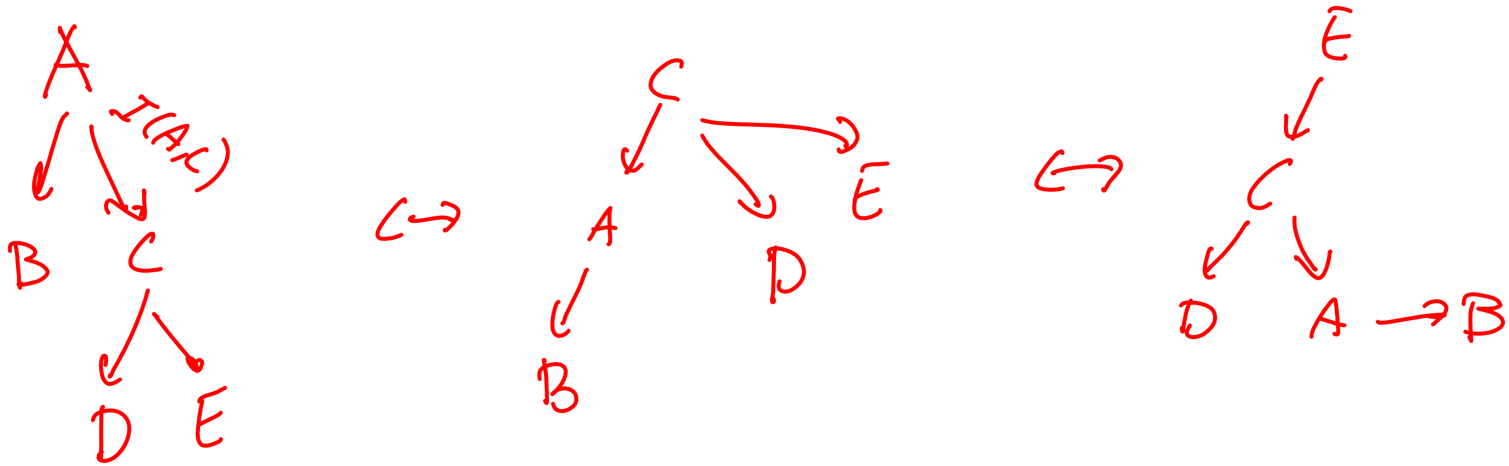
for n vars

$\omega(n!)$

Scoring a tree 1: I-equivalent trees

forget

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \cancel{\sum_i \hat{H}(X_i)}$$



$I(A,B) + I(A,C) + I(C,D) + I(C,E)$ $=$ Same $=$ Same

Scoring a tree 2: similar trees

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \hat{H}(X_i)$$

$$\begin{tikzcd}
 & A \arrow[dl] \arrow[dr] & \\
B & & C \arrow[d, "I(C,D)"] \\
 & & D
\end{tikzcd}$$

$I(A,B) + I(A,C) + I(C,D)$

$$\begin{tikzcd}
 & A \arrow[dl] \arrow[d] \arrow[dr, "I(A,D)"] & \\
B & C & D
\end{tikzcd}$$

$I(A,B) + I(A,C) + I(A,D)$

Chow-Liu tree learning algorithm 1

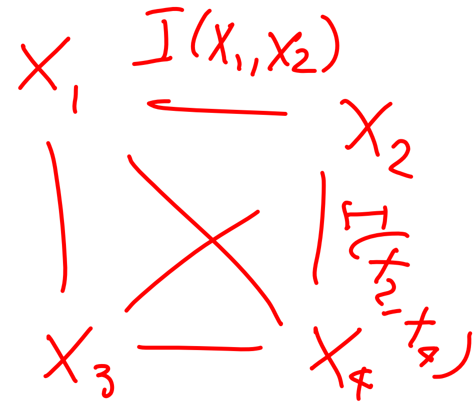
- For each pair of variables X_i, X_j
  - Compute empirical distribution:

$$\widehat{P}(x_i, x_j) = \frac{\text{Count}(x_i, x_j)}{M}$$

  - Compute mutual information:

$$\widehat{I}(X_i, X_j) = \sum_{x_i, x_j} \widehat{P}(x_i, x_j) \log \frac{\widehat{P}(x_i, x_j)}{\widehat{P}(x_i)\widehat{P}(x_j)}$$

- Define a graph
  - Nodes $X_1, \ldots, X_n$
  - Edge (i,j) gets weight $\widehat{I}(X_i, X_j)$

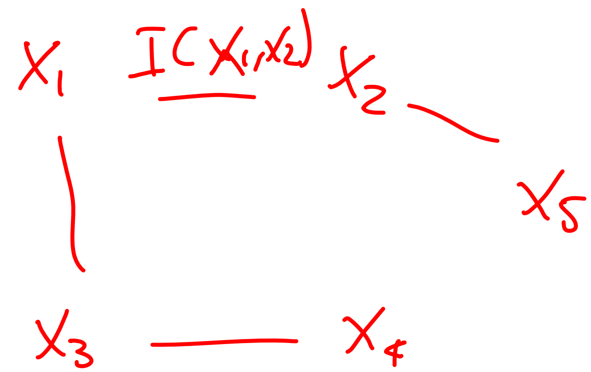


Chow-Liu tree learning algorithm 2

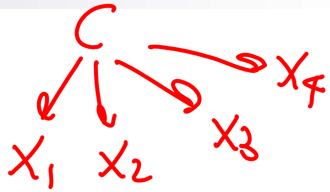
$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \hat{H}(X_i)$$

- Optimal tree BN
  - Compute maximum weight spanning tree
  - Directions in BN: pick any node as root, breadth-first-search defines directions

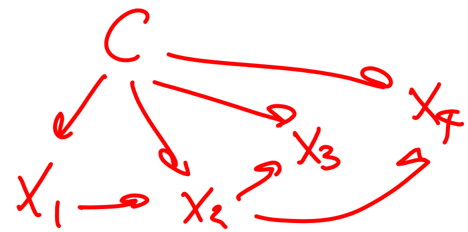
because of I-equivalence



Can we extend Chow-Liu 1



- Tree augmented naïve Bayes (TAN) [Friedman et al. '97]
  - Naïve Bayes model overcounts, because correlation between features not considered
  - Same as Chow-Liu, but score edges with:

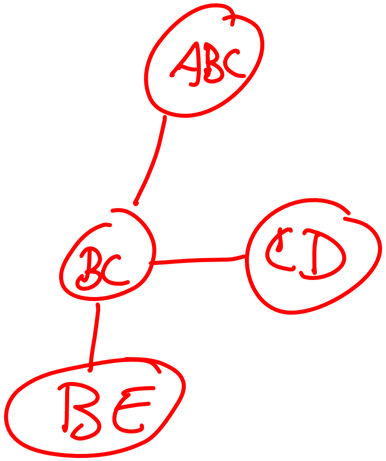


$$\hat{I}(X_i, X_j \mid C) = \sum_{c, x_i, x_j} \hat{P}(c, x_i, x_j) \log \frac{\hat{P}(x_i, x_j \mid c)}{\hat{P}(x_i \mid c)\hat{P}(x_j \mid c)}$$

$$\text{Score}(\text{Tan}) = \sum_i I(C, X_i) + \sum_j I(X_j, \{Pa_{X_j}, C\})$$

Can we extend Chow-Liu 2

- (Approximately learning) models with tree-width up to *k*
  - [Narasimhan & Bilmes ’04]
  - But, $O(n^{k+1})$…



Maximum likelihood overfits!

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = M \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \hat{H}(X_i)$$

- Information never hurts:

$$\begin{tikzcd} A \arrow[d] & C \\ B & \end{tikzcd} \dashrightarrow \begin{tikzcd} A \arrow[dr] & & C \arrow[dl] \\ & B & \end{tikzcd}$$

$$I(B,A) \leq I(B,\{A,C\})$$

- Adding a parent always increases score!!!

Fully connected graph wins!

Bayesian score

- Prior distributions:
  - Over structures $P(G)$
  - Over parameters of a structure $P(\theta_G \mid G)$
- Posterior over structures given data:

$$\log P(G \mid D) \overset{K+}{\propto} \log P(D \mid G) \cdot P(G) = \log P(G) + \overbrace{\log P(D \mid G)}$$

$$P(D \mid G) = \int_{\theta_G} P(D, \theta_G \mid G)\, d\theta_G = \int_{\theta_G} \underbrace{P(D \mid \theta_G, G)}_{\text{likelihood}} \cdot \underbrace{P(\theta_G \mid G)}_{\text{prior}}\, d\theta_G$$

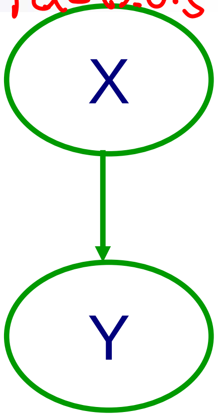
$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} \mid \mathcal{G}) d\theta_{\mathcal{G}}$$

Bayesian score and model complexity

$P(\theta_X|G) \cdot P(\theta_Y|G)$ param. indep

$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_\mathcal{G}} P(D \mid \mathcal{G}, \theta_\mathcal{G}) P(\theta_\mathcal{G}|\mathcal{G}) d\theta_\mathcal{G}$$

True model: $P(X=t)=0.5$



- Structure 1: X and Y independent X Y

$$\log P(D|G) = \log \int_{\theta_X} \int_{\theta_Y} P(D_X \mid \theta_X) \cdot P(\theta_X) \cdot P(D_Y|\theta_Y) P(\theta_Y) d\theta_X d\theta_Y$$

$$= \log \int_{\theta_X} P(D_X|\theta_X) P(\theta_X) d\theta_X + \log \int_{\theta_Y} P(D_Y|\theta_Y) P(\theta_Y) d\theta_Y$$

  - Score doesn't depend on alpha
- Structure 2: X→Y

$$\log P(D|G) = \log \int_{\theta_X} P(D_X|\theta_X) P(\theta_X) d\theta_X +$$

$$+ \log \int_{\theta_{Y|X=t}} P(D_{Y|X=t} \mid \theta_{Y|X=t}) P(\theta_{X=t}|G) d\theta_{Y|X=t} + \int_{\theta_{Y|X=f}} \cdots$$

P(Y=t|X=t) = 0.5 + α
P(Y=t|X=f) = 0.5 - α

  - Data points split between P(Y=t|X=t) and P(Y=t|X=f)
  - For fixed M, only worth it for large α
    - Because posterior of less diffuse

Bayesian, a decomposable score

$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_\mathcal{G}} P(D \mid \mathcal{G}, \theta_\mathcal{G}) P(\theta_\mathcal{G} | \mathcal{G}) d\theta_\mathcal{G}$$

- As with last lecture, assume:
  - Local and global parameter independence $(\theta_{x=t} \perp \theta_{y=t})$
- Also, prior satisfies **parameter modularity**:
  - If X_i has same parents in *G* and *G'*, then parameters have same prior

$Pa_{x_i,G} = Pa_{x_i,G'} = U \Rightarrow P(\theta_{x_i|Pa_{x_i}}|G) = P(\theta_{x_i|Pa_{x_i}} | G')$

- Finally, structure prior P(*G*) satisfies **structure modularity**
  - Product of terms over families
  - E.g., P(*G*) $\propto c^{|G|}$

- Bayesian score decomposes along families!

BIC approximation of Bayesian score

- Bayesian has difficult integrals
- For Dirichlet prior, can use simple Bayes information criterion (BIC) approximation
  - In the limit, we can forget prior!
  - **Theorem**: for Dirichlet prior, and a BN with Dim(*G*) independent parameters, as M$\rightarrow\infty$:

$$\log P(D \mid \mathcal{G}) = \log \underbrace{P(D \mid \mathcal{G}, \theta_{\mathcal{G}})}_{\text{likelihood score}} - \frac{\log M}{2} \underbrace{\mathrm{Dim}(\mathcal{G})}_{\text{get worse as complicated}} + O(1)$$

BIC approximation, a decomposable score

MAP $\theta_{x|y} = \frac{\text{Count}(x + \alpha_x, y + \alpha_y)}{\text{normalize}}$

- BIC: $$\text{Score}_{\text{BIC}}(\mathcal{G} : D) = \log P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) - \frac{\log M}{2} \text{Dim}(\mathcal{G})$$

- Using information theoretic formulation:

$$\text{Score}_{\text{BIC}}(\mathcal{G} : D) = M \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - M \sum_i \hat{H}(X_i) - \frac{\log M}{2} \sum_i \text{Dim}(P(X_i \mid \mathbf{Pa}_{x_i,\mathcal{G}}))$$

$-c \cdot |G|$

Introduce $P(G)$:

$P(G) \propto c^{-|G|}$

$\log P(G) = -c \cdot |G| + K$

Consistency of BIC and Bayesian scores

Consistency is limiting behavior, says nothing about finite sample size!!!

- A scoring function is **consistent** if, for true model G^*, as $M \to \infty$, with probability 1
  - G^* maximizes the score
  - All structures **not I-equivalent** to G^* have strictly lower score
- **Theorem**: BIC score is consistent
- **Corollary**: the Bayesian score is consistent
- What about maximum likelihood?

G^*: X Y ; X → Y

same likelihood score

cond. 2 violated

Priors for general graphs

- For finite datasets, prior is important!
- Prior over structure satisfying prior modularity

- What about prior over parameters, how do we represent it?
  - *K2 prior*: fix an α, $P(\theta_{X_i|\mathbf{Pa}_{X_i}}) = \text{Dirichlet}(\alpha, \ldots, \alpha)$
  - K2 is "inconsistent"

BDe prior

- Remember that Dirichlet parameters analogous to “fictitious samples”
- Pick a fictitious sample size M’
- For each possible family, define a prior distribution $P(X_i,\mathbf{Pa}_{X_i})$
  - Represent with a BN
  - Usually independent (product of marginals)
- **BDe prior**:
- Has “consistency property”:

Score equivalence

- If *G* and *G'* are I-equivalent then they have same score

- **Theorem**: Maximum likelihood and BIC scores satisfy score equivalence
- **Theorem**:
  - If P(*G*) assigns same prior to I-equivalent structures (e.g., edge counting)
  - and parameter prior is dirichlet
  - then Bayesian score satisfies score equivalence if and only if prior over parameters represented as a BDe prior!!!!!!

Chow-Liu for Bayesian score

- Edge weight $w_{X_j \to X_i}$ is advantage of adding X_j as parent for X_i

- Now have a directed graph, need directed spanning forest
  - Note that adding an edge can hurt Bayesian score – choose forest not tree
  - But, if score satisfies score equivalence, then $w_{X_j \to X_i} = w_{X_j \to X_i}$!
  - Simple maximum spanning forest algorithm works

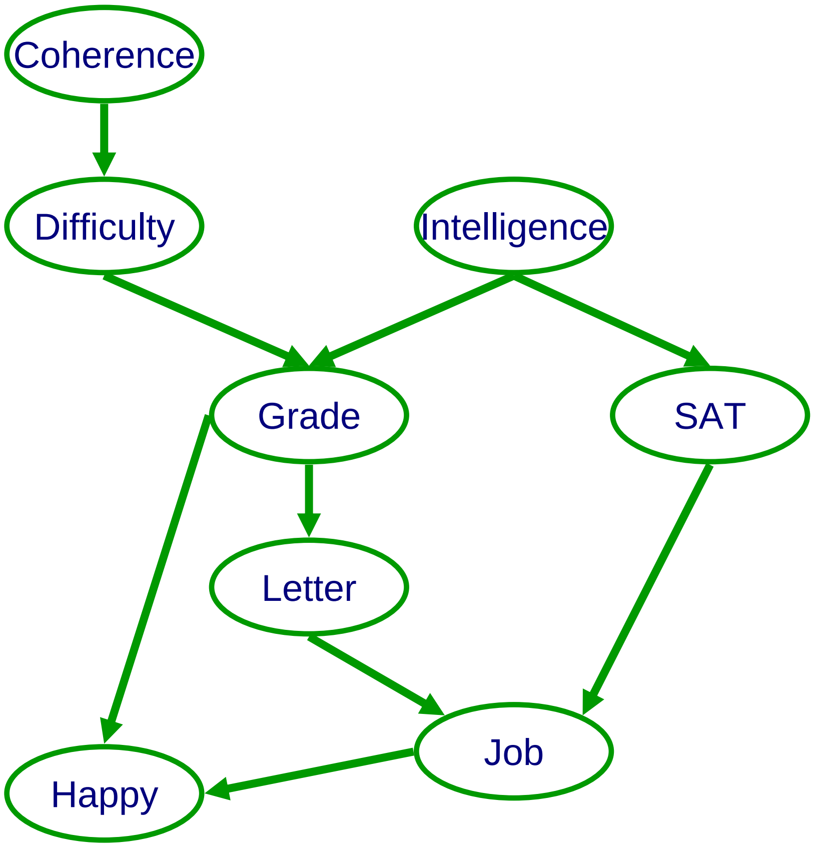
Structure learning for general graphs

- In a tree, a node only has one parent

- **Theorem**:
  - The problem of learning a BN structure with at most d parents is NP-hard for any (fixed) $d \geq 2$

- Most structure learning approaches use heuristics
  - Exploit score decomposition
  - (Quickly) Describe two heuristics that exploit decomposition in different ways

Understanding score decomposition



Fixed variable order 1

- Pick a variable order $\prec$
  - e.g., $X_1,\ldots,X_n$
- X_i can only pick parents in $\{X_1,\ldots,X_{i-1}\}$
  - Any subset
  - Acyclicity guaranteed!
- Total score = sum score of each node

Fixed variable order 2

- Fix max number of parents to k
- For each *i* in order $\prec$
  - Pick $\mathbf{Pa}_{X_i} \subseteq \{X_1,\ldots,X_{i-1}\}$
    - Exhaustively search through all possible subsets
    - $\mathbf{Pa}_{X_i}$ is maximum $\mathbf{U} \subseteq \{X_1,\ldots,X_{i-1}\}$ FamScore($X_i | \mathbf{U} : D$)

- Optimal BN for each order!!!
- Greedy search through space of orders:
  - E.g., try switching pairs of variables in order
  - If neighboring vars in order are switch, only need to recompute score for this pair
    - O(n) speed up per iteration
    - Local moves may be worse

Learn BN structure using local search

Starting from Chow-Liu tree

Local search,

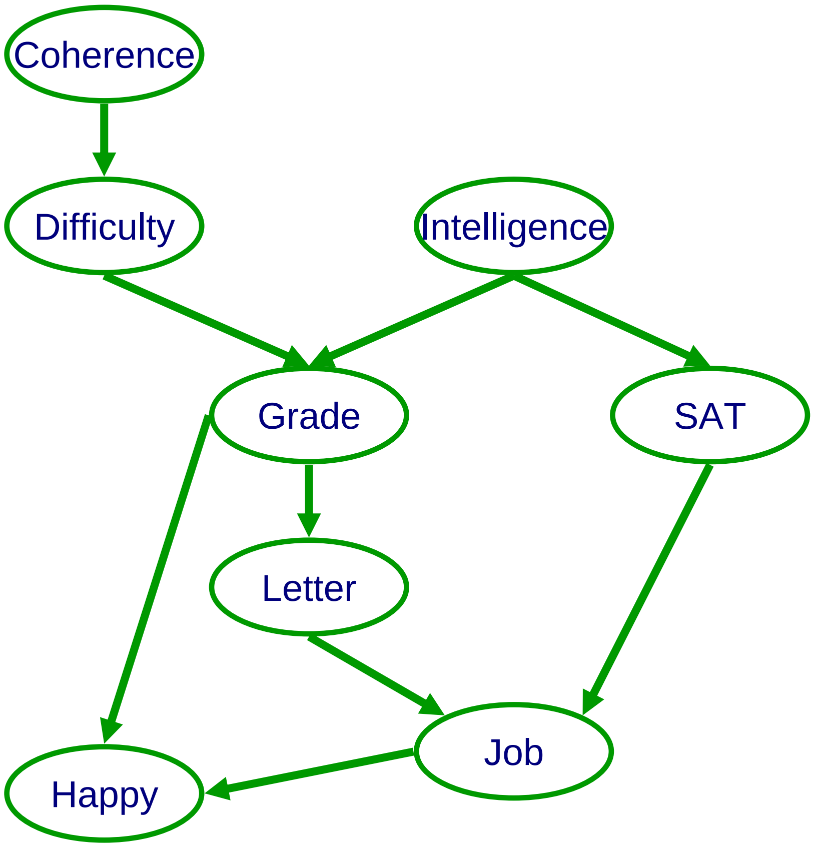
possible moves:

Only if acyclic!!!

- Add edge
- Delete edge
- Invert edge

Select using favorite score

Exploit score decomposition in local search



- Add edge and delete edge:
  - Only rescore one family!
- Reverse edge
  - Rescore only two families

Order search versus graph search

- Order search advantages
  - For fixed order, optimal BN – more “global” optimization
  - Space of orders much smaller than space of graphs

- Graph search advantages
  - Not restricted to k parents
    - Especially if exploiting CPD structure, such as CSI
  - Cheaper per iteration
  - Finer moves within a graph

Bayesian model averaging

- So far, we have selected a single structure
- But, if you are really Bayesian, must average over structures
  - Similar to averaging over parameters

$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} | \mathcal{G}) d\theta_{\mathcal{G}}$$

- Inference for structure averaging is very hard!!!
  - Clever tricks in reading

What you need to know about learning BN structures

- Decomposable scores
  - Maximum likelihood
  - Information theoretic interpretation
  - Bayesian
  - BIC approximation
- Priors
  - Structure and parameter assumptions
  - BDe if and only if score equivalence
- Best tree (Chow-Liu)
- Best TAN
- Nearly best k-treewidth (in $O(N^{k+1})$)
- Search techniques
  - Search through orders
  - Search through structures
- Bayesian model averaging