Stat 521A
Lecture 24

Outline

- Scoring functions for DAGs with hidden vars (19.4.1)
- Structure search (19.4.2)
- Structural EM (19.4.3)
- Inventing hidden variables in DGMs (19.5)

Bayesian score

- Need a way to measure model quality; orthogonal to issue of how we search through space of models
- Bayesian score hard to compute since posterior is an exponential number of modes

$$\text{score}_B(\mathcal{G} : \mathcal{D}) = \log P(\mathcal{D} \mid \mathcal{G}) + \log P(\mathcal{G})$$

$$\begin{aligned} p(D|G) &= \int \prod_m p(\mathbf{o}[m]|\boldsymbol{\theta}, G) p(\boldsymbol{\theta}|G) d\boldsymbol{\theta} \\ p(\mathbf{o}[m]|\boldsymbol{\theta}, G) &= \sum_{\mathbf{h}} p(\mathbf{o}[m], \mathbf{h}|\boldsymbol{\theta}, G) \end{aligned}$$

- Approximations: asymptotic, variational, MCMC

Chib's candidate method

- Approximate p(D|G) using output of a standard MCMC run. For any θ (eg MAP) compute

$$P(\mathcal{D} \mid \mathcal{G}) = \frac{P(\mathcal{D} \mid \theta, \mathcal{G}) P(\theta \mid \mathcal{G})}{P(\theta \mid \mathcal{D}, \mathcal{G})}.$$

- Requires that p(θ|D,G) cover chosen θ.
- This requires that MCMC mix over all posterior modes, even if symmetrical. If not, it will underestimate p(D|G). See rejected letter to editor by Radford Neal.*

* http://www.cs.utoronto.ca/~radford/ftp/chib-letter.pdf

RJMCMC

- Instead of doing discrete search, and integrating out params at each point, let us jointly sample in graph and param space
- Since the size of the cts space is changing, we need to use a change of measure when we move between dimensionalities
- This results in reversible jump MCMC
- Getting it working is delicate…

Laplace approximation

Box 19.F — Concept: Laplace Approximation. *The Laplace approximation can be applied to any function of the form $f(\boldsymbol{w}) = e^{g(\boldsymbol{w})}$ for some vector $\boldsymbol{w}$. Our task is to compute the integral*

$$F = \int f(\boldsymbol{w})d\boldsymbol{w}$$

Using Taylor's expansion, we can expand an approximation of g around a point $\boldsymbol{w}_0$

$$g(\boldsymbol{w}) \approx g(\boldsymbol{w}_0) + \left[\frac{\partial g(\boldsymbol{w})}{\partial x_i}\right] |_{\boldsymbol{w}=\boldsymbol{w}_0} (\boldsymbol{w} - \boldsymbol{w}_0) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_0)^T \left[\frac{\partial\partial g(\boldsymbol{w})}{\partial x_i \partial x_j}\right] |_{\boldsymbol{w}=\boldsymbol{w}_0} (\boldsymbol{w} - \boldsymbol{w}_0).$$

where $\left[\frac{\partial g(\boldsymbol{w})}{\partial x_i}\right] |_{\boldsymbol{w}=\boldsymbol{w}_0}$ denotes the vector of first derivatives and $\left[\frac{\partial\partial g(\boldsymbol{w})}{\partial x_i \partial x_j}\right] |_{\boldsymbol{w}=\boldsymbol{w}_0}$ denotes the Hessian *— the matrix of second derivatives.*

If $\boldsymbol{w}_0$ is the maximum of $g(\boldsymbol{w})$, then the second term disappears. We now set

$$\boldsymbol{C} = -\left[\frac{\partial^2 g(\boldsymbol{w})}{\partial x_i \partial x_j}\right] |_{\boldsymbol{w}=\boldsymbol{w}_0}$$

to be the negative of the matrix of second derivatives of $g(\boldsymbol{w})$ at $\boldsymbol{w}_0$. Since $\boldsymbol{w}_0$ is a maximum, this matrix is positive semi-definitive. Thus, we get the approximation

$$g(\boldsymbol{w}) \approx g(\boldsymbol{w}_0) - \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_0)^T \boldsymbol{C}(\boldsymbol{w} - \boldsymbol{w}_0).$$

Plugging this approximation into the definition of $f(x)$, we can write

$$\int f(\boldsymbol{w})d\boldsymbol{w} \approx f(\boldsymbol{w}_0) \int e^{-\frac{1}{2}(\boldsymbol{w}-\boldsymbol{w}_0)^T \boldsymbol{C}(\boldsymbol{w}-\boldsymbol{w}_0)} d\boldsymbol{w}.$$

The integral is identical to the integral of an unnormalized Gaussian distribution with covariance matrix $\Sigma = C^{-1}$. We can therefore solve this integral analytically and obtain:

$$\int f(\boldsymbol{w})d\boldsymbol{w} \approx f(\boldsymbol{w}_0)|\boldsymbol{C}|^{-\frac{1}{2}}(2\pi)^{\frac{1}{2}\dim(\boldsymbol{C})}$$

where $\dim(\boldsymbol{C})$ is the dimension of the matrix $\boldsymbol{C}$.

Laplace approximation cont'd

- Let g(w) = log p(D,w|G).
- Laplace approximation to p(D,G) is

$$\text{score}_{Laplace}(\mathcal{G} : \mathcal{D}) = \log P(\mathcal{G}) + \log P(\mathcal{D} \mid \tilde{\theta}_{\mathcal{G}}, \mathcal{G}) + \frac{\dim(C)}{2} \log 2\pi - \frac{1}{2} \log |C|,$$

C is negative Hessian: requires inference on xi, xj, ui, uj

$$-\left.\frac{\partial^2 \log P(\mathcal{D} \mid \theta, \mathcal{G})}{\partial \theta_{x_i|\boldsymbol{u}_i} \partial \theta_{x_j|\boldsymbol{u}_j}}\right|_{\tilde{\boldsymbol{\theta}}_{\mathcal{G}}} = -\sum_m \left.\frac{\partial^2 \log P(\boldsymbol{o}[m] \mid \theta, \mathcal{G})}{\partial \theta_{x_i|\boldsymbol{u}_i} \partial \theta_{x_j|\boldsymbol{u}_j}}\right|_{\tilde{\boldsymbol{\theta}}_{\mathcal{G}}},$$

BIC score

- BIC is the limit of Laplace as M->inf.

$$C = \sum_{m=1}^{M} C_m \qquad C = M \frac{1}{M} \sum_{m=1}^{M} C_m.$$

$$\det(C) = M^{\dim(C)} \det\left(\frac{1}{M}\sum_{m=1}^{M} C_m\right) \approx M^{\dim(C)} \det\left(E_{P^*}[C_o]\right).$$

$$\log\det(C) \approx \dim(C)\log M + \log\det\left(E_{P^*}[C_o]\right).$$

Theorem 19.4.1: *As $M \to \infty$, we have that:*

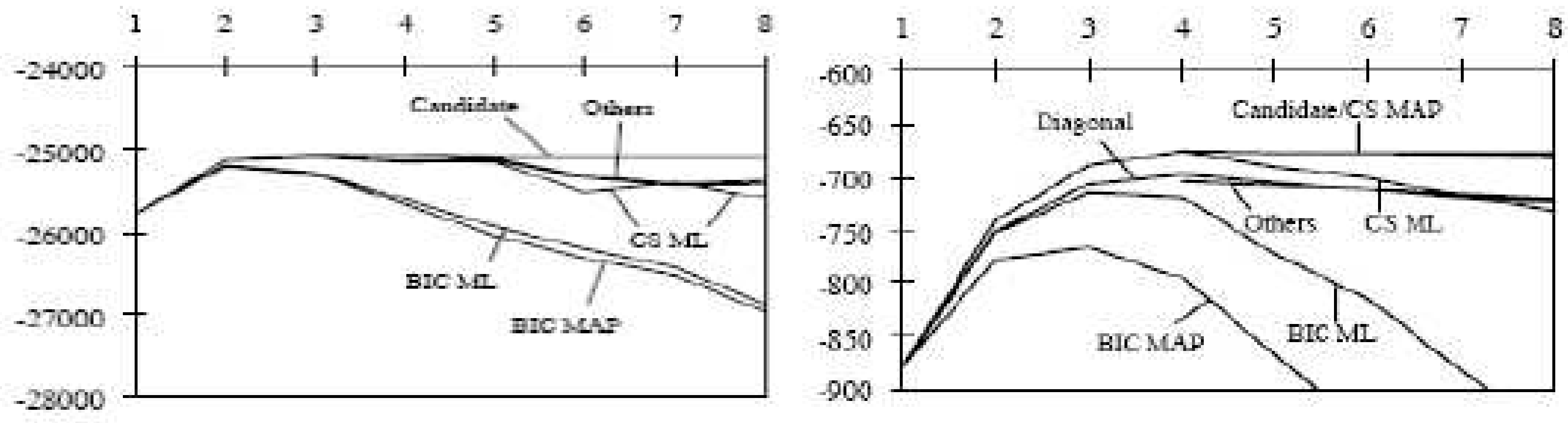
$$\text{score}_{Laplace}(\mathcal{G} : \mathcal{D}) = \text{score}_{BIC}(\mathcal{G} : \mathcal{D}) + O(1)$$

where $\text{score}_{BIC}(\mathcal{G} : \mathcal{D})$ *is the BIC score*

$$\text{score}_{BIC}(\mathcal{G} : \mathcal{D}) = \log P(\mathcal{D} \mid \tilde{\theta}_{\mathcal{G}}, \mathcal{G}) - \frac{\log M}{2}\text{Dim}[\mathcal{G}] + \log P(\mathcal{G}) + \log P(\tilde{\theta}_{\mathcal{G}} \mid \mathcal{G}).$$

Cheeseman-Stutz approximation

- CS approx to log p(D|G) is more accurate than BIC, yet faster than Laplace
- Matt Beal's thesis proves CS is a lower bound
- Example: we plot log p(D|K) vs K for a mixture of Bernoullis for different methods; 'candidate' is a 'gold standard' MCMC method



CS approx

- Idea 1: If D* is complete, p(D*|G) just relies on sufficient statistics, so use ESS instead

$$P(\mathcal{D}^*_{\mathcal{G},\tilde{\theta}_\mathcal{G}} \mid \mathcal{G}) = \int p(\mathcal{D}^*_{\mathcal{G},\tilde{\theta}_\mathcal{G}} \mid \theta, \mathcal{G}) P(\theta \mid \mathcal{G}) d\theta$$

- Unfortunately this does not work well, since it sums over 1 (imputed) dataset whereas p(D|G) sums over an exponential number

$$P(\mathcal{D} \mid \mathcal{G}) = \int \sum_{\mathcal{H}} p(\mathcal{D}, \mathcal{H} \mid \theta, \mathcal{G}) P(\theta \mid \mathcal{G}) d\theta = \sum_{\mathcal{H}} \int p(\mathcal{D}, \mathcal{H} \mid \theta, \mathcal{G}) P(\theta \mid \mathcal{G}), d\theta.$$

- Idea 2: add an approximate correction term

$$\log P(\mathcal{D} \mid \mathcal{G}) \quad = \quad \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\theta}_\mathcal{G}} \mid \mathcal{G}) + \log P(\mathcal{D} \mid \mathcal{G}) - \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\theta}_\mathcal{G}} \mid \mathcal{G})$$

Approximate with BIC

CS approx

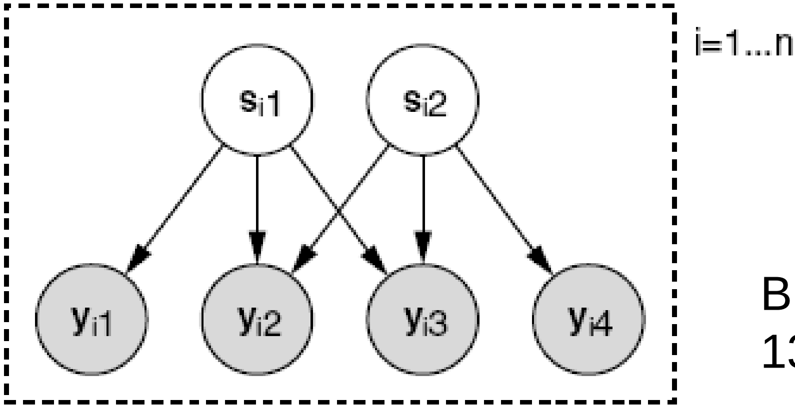
$$\log P(\mathcal{D} \mid \mathcal{G}) - \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \mathcal{G}) \approx \left[\log P(\mathcal{D} \mid \tilde{\theta}_\mathcal{G}, \mathcal{G}) - \frac{1}{2}\mathrm{Dim}[\tilde{\theta}_\mathcal{G}] \log M\right] - \left[\log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \tilde{\theta}_\mathcal{G}, \mathcal{G}) - \frac{1}{2}\mathrm{Dim}[\tilde{\theta}_\mathcal{G}] \log M\right] = \log P(\mathcal{D} \mid \tilde{\theta}_\mathcal{G}, \mathcal{G}) - \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \tilde{\theta}_\mathcal{G}, \mathcal{G}).$$

$$\log P(\mathcal{D} \mid \mathcal{G}) = \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \mathcal{G}) + \log P(\mathcal{D} \mid \mathcal{G}) - \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \mathcal{G}) \approx \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \mathcal{G}) + \log P(\mathcal{D} \mid \tilde{\theta}_\mathcal{G}, \mathcal{G}) - \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \tilde{\theta}_\mathcal{G}, \mathcal{G}).$$

$$\mathrm{score}_{CS}(\mathcal{G} : \mathcal{D}) = \log P(\mathcal{G}) + \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \mathcal{G}) + \log P(\mathcal{D} \mid \tilde{\theta}_\mathcal{G}, \mathcal{G}) - \log P(\mathcal{D}^*_{\mathcal{G},\tilde{\boldsymbol{\theta}}_\mathcal{G}} \mid \tilde{\theta}_\mathcal{G}, \mathcal{G})$$

Variational lower bound

EM for MAP estimation	Variational Bayesian EM
Goal: maximise $p(\theta \mid \mathbf{y}, m)$ w.r.t. θ **E Step:** compute $q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) = p(\mathbf{x} \mid \mathbf{y}, \theta^{(t)})$ **M Step:** $\theta^{(t+1)} = \arg\max_{\theta} \int d\mathbf{x}\, q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) \ln p(\mathbf{x}, \mathbf{y}, \theta)$	**Goal:** lower bound $p(\mathbf{y} \mid m)$ **VBE Step:** compute $q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) = p(\mathbf{x} \mid \mathbf{y}, \overline{\phi}^{(t)})$ **VBM Step:** $q_{\theta}^{(t+1)}(\theta) \propto \exp \int d\mathbf{x}\, q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) \ln p(\mathbf{x}, \mathbf{y}, \theta)$

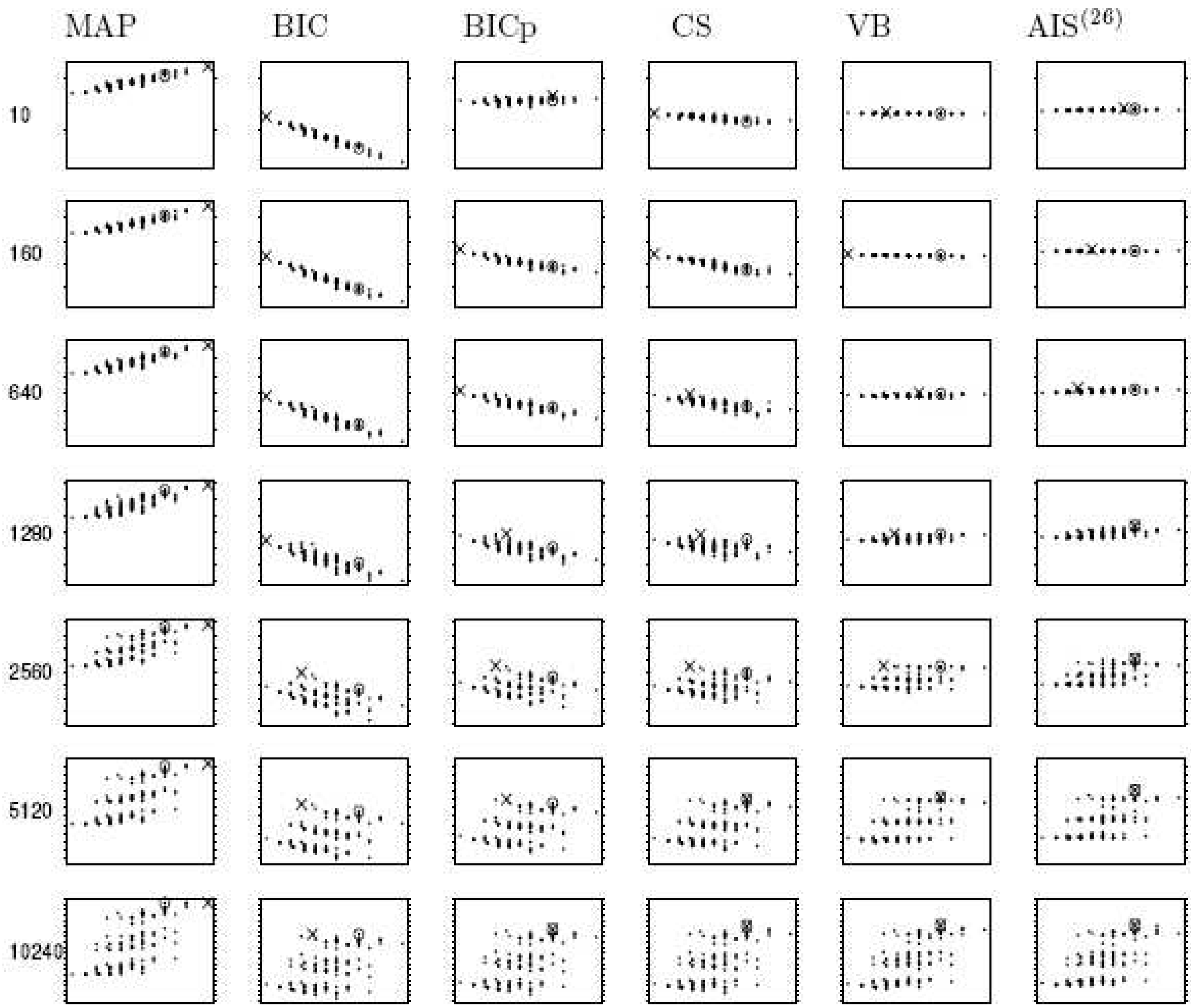


VB provably tighter lower
Bound than CS

Binary hidden nodes, 5-ary obs nodes
136 distinct DAGs

Beal, M.J. and Ghahramani, Z.
Variational Bayesian Learning of Directed Graphical Models with Hidden Variables
Bayesian Analysis **1**(4), 2006.

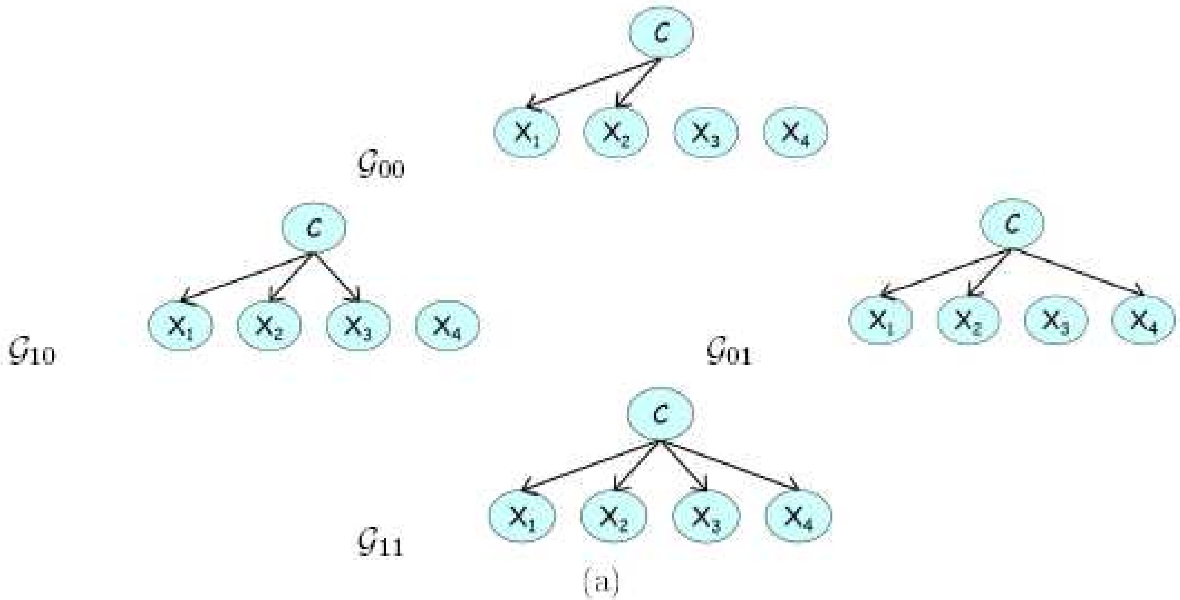
Log p(D|G) vs dof(G)



Structure search

- P(D|G) does not factorize across families, unlike the fully observed case
- Cannot find (easily) optimal tree or optimal DAG given ordering.
- For local search, evaluating score of neighbors is expensive – score does not decompose, so need to find MAP estimate for each graph just to compute its BIC score

Illustration of non-decomposability



{1,2} and 3: weak corr
3 and 4: strong corr

Network	ΔLL	ΔCS
$\mathcal{G}_{10}$ (add $C \rightarrow X_3$)	+3	−0.4
$\mathcal{G}_{01}$ (add $C \rightarrow X_4$)	+10.6	+7.2
$\mathcal{G}_{11}$ (add $C \rightarrow X_3$, $C \rightarrow X_4$)	+24.1	+17.4

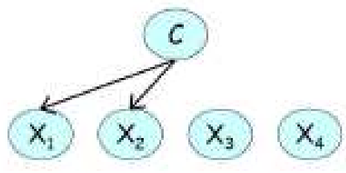
Structural EM

- Given current graph Gt and MAP params theta(t), compute ESS for all possible families (potentially in a lazy fashion – may need out-of-clq queries)
- Evaluate BIC score for G(t+1) using ESS|G(t)
- Thm: increasing expected BIC score increases true BIC score

Theorem 19.4.3: *Let $\mathcal{G}_0$ be a graph structure and $\tilde{\theta}_0$ be the MAP parameters for $\mathcal{G}_0$ given a dataset $\mathcal{D}$. Then for any graph structure $\mathcal{G}$:*

$$\text{score}_{BIC}(\mathcal{G} : \mathcal{D}^*_{\mathcal{G}_0,\tilde{\boldsymbol{\theta}}_0}) - \text{score}_{BIC}(\mathcal{G}_0 : \mathcal{D}^*_{\mathcal{G}_0,\tilde{\boldsymbol{\theta}}_0}) \leq \text{score}_{BIC}(\mathcal{G} : \mathcal{D}) - \text{score}_{BIC}(\mathcal{G}_0 : \mathcal{D}).$$

Sparse mixture model



$\mathcal{G}_{00}$

- Run parameter estimation (such as EM or gradient ascent) to learn parameters $\tilde{\boldsymbol{\theta}}_t$ for $\mathcal{G}_t$.
- Construct a new structure $\mathcal{G}_{t+1}$ so that $\mathcal{G}_{t+1}$ contains the edge $C \rightarrow X_i$ if

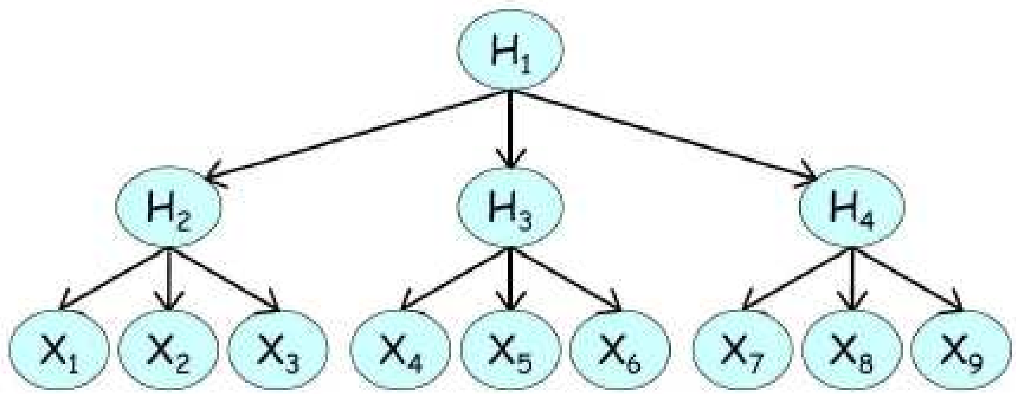
$$\text{FamScore}(X_i, \{C\} \,:\, \mathcal{D}^*_{\mathcal{G}_t, \tilde{\boldsymbol{\theta}}_t}) > \text{FamScore}(X_i, \emptyset \,:\, \mathcal{D}^*_{\mathcal{G}_t, \tilde{\boldsymbol{\theta}}_t}).$$

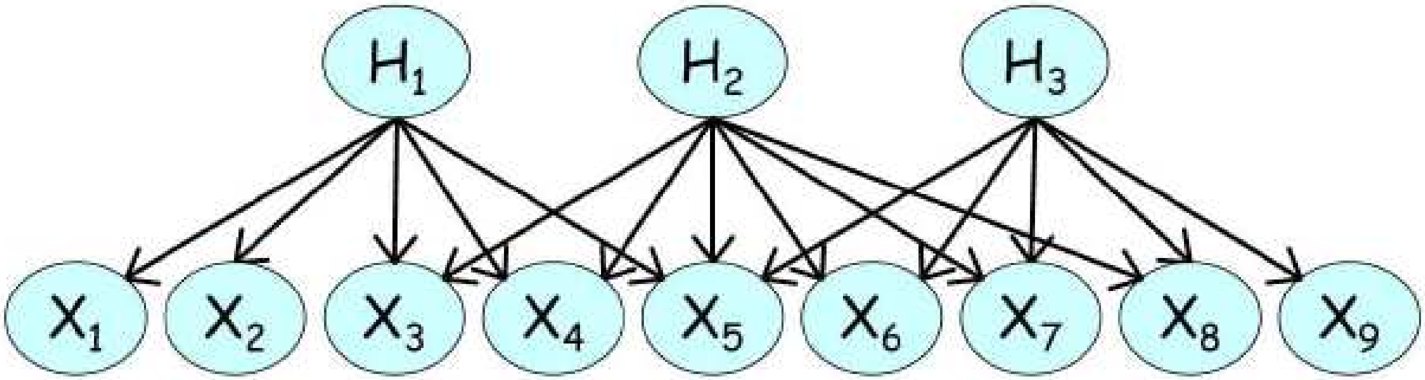
$$\begin{aligned} \bar{M}_{\mathcal{D}^*_{\mathcal{G}_t, \tilde{\boldsymbol{\theta}}_t}}[x_i, c] &= \sum_m P(C[m] = c, X_i[m] = x_i \mid \mathbf{o}[m], \mathcal{G}_t, \tilde{\boldsymbol{\theta}}_t) \\ &= \sum_{m, X_i[m] = x_i} P(C[m] = c \mid \mathbf{o}[m], \mathcal{G}_t, \tilde{\boldsymbol{\theta}}_t). \end{aligned}$$

Initialization: if start from no children, will never add any! So start from all Children or random subset.

Inventing hidden variables

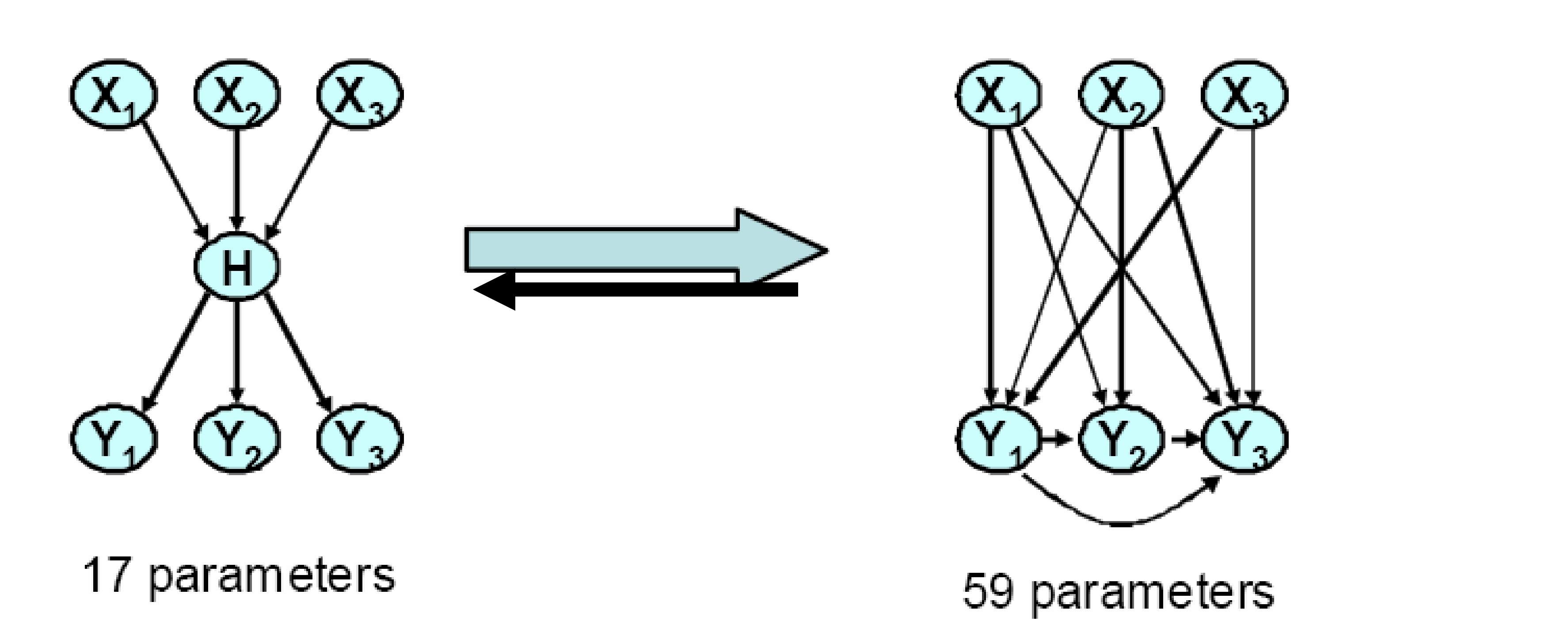
- Can add hidden variables in 'canonical' places





Structural signatures

- Can learn structure with no hidden vars, then look for 'semi-cliques'.
- Unfortunately original model discourages nodes with high fan-in.

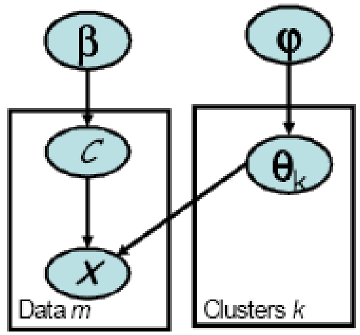


Can also look for signatures in the data - eg FCI* algorithm

Cardinality of hidden nodes

- Need to choose number of states.
- Can use an “infinite” number using Dirichlet processes.
- Let us first consider DP mixture models.

Marginalizing out θ



Collapsed Gibbs sampling
Cf DP mixtures

$$P(\boldsymbol{\lambda} \mid \boldsymbol{c}) = Dirichlet(\alpha_0/K + |I_1(\boldsymbol{c})|, \ldots, \alpha_0/K + |I_K(\boldsymbol{c})|).$$

$$P(\boldsymbol{\theta}_k \mid \boldsymbol{c}, \mathcal{D}, \phi) = Q(\boldsymbol{\theta}_k \mid \mathcal{D}_{I_k(\boldsymbol{c})}, \phi) \propto P(\boldsymbol{\theta}_k \mid \phi) \prod_{m \in I_k(\boldsymbol{c})} P(\boldsymbol{x}[m] \mid \boldsymbol{\theta}_k),$$

$$P(C[m'] = k \mid \boldsymbol{c}_{-m'}, \mathcal{D}, \phi) \propto P(C[m'] = k \mid \boldsymbol{\lambda}, \boldsymbol{c}_{-m'}) P(\boldsymbol{x}[m'] \mid C[m'] = k, \boldsymbol{x}[I_k(\boldsymbol{c}_{-m'})], \phi).$$

$$P(C[m'] = k \mid \boldsymbol{c}_{-m'}, \mathcal{D}, \phi) \propto (|I_k(\boldsymbol{c}_{-m'})| + \alpha_0/K) Q(\boldsymbol{X} \mid \mathcal{D}_{I_k(\boldsymbol{c}_{-m'})}, \phi).$$

O(M K) per iter

DP mixture model (p865)

- Identity of clusters does not matter. Let σ={I1,...,IL} be a partition, Ic=cases in cluster c. For case m', either join existing cluster or create new one O(ML) per iter

$$P(I \leftarrow I \cup \{m'\} \mid \sigma_{-m'}, \mathcal{D}, \phi) \propto \left(|I| + \frac{\alpha_0}{K}\right) Q(x[m'] \mid \mathcal{D}_I, \phi)$$
$$P(\sigma \leftarrow \sigma \cup \{\{m'\}\} \mid \sigma_{-m'}, \mathcal{D}, \phi) \propto (K - L)\frac{\alpha_0}{K} Q(x[m'] \mid \phi),$$

- Now let K->inf.

$$P(I \leftarrow I \cup \{m'\} \mid \sigma_{-m'}, \mathcal{D}, \phi) \propto |I| \cdot Q(x[m'] \mid \mathcal{D}_I, \phi)$$
$$P(\sigma \leftarrow \sigma \cup \{\{m'\}\} \mid \sigma_{-m'}, \mathcal{D}, \phi) \propto \alpha_0 \cdot Q(x[m'] \mid \phi).$$

- More likely to join a cluster if it is already crowded.
- Chinese Restaurant process.