Learning the Structure of Probabilistic Models

Nir Friedman

U.C. Berkeley

Parts of this work were done while at SRI International

1

Graphical Models

Method	Applications
Hidden Markov Models	Speech recognition
Mixture Models	Star catalog, market analysis, digit recognition
Bayesian Networks	Causal models, sensor fusion, expert systems
Markov Random Fields	Vision
Linear Structural Eq.	Econometeric models, social sciences
Phylogenetic Trees	Evolution

2

Learning in Graphical Models

Structure: captures dependencies between random variables in a graphical model.

- Example: whether smoking is related to lung cancer.

A fixed structure defines a parametric family of distributions.

Central Tool: Expectation Maximization (EM)

- Allows to optimize choice of parameters for a fixed structure.

3

Learning structure in probabilistic models

I will describe, Structural EM, a new search technique that

- combines parametric methods (EM) with a search over structures
- results in effective learning procedures
- easily applies to learning Bayesian networks
- applies to other families of probabilistic models as well.

4

Outline

- Introduction to Bayesian networks.
- Learning Bayesian networks from complete data.
- Learning Bayesian networks from incomplete data.
- The Structural EM algorithm.
- Experimental results.
- Future directions and discussion.

5

Bayesian Networks

The method of choice for representing uncertainty in AI.

Utilize explicit representation of structure to:

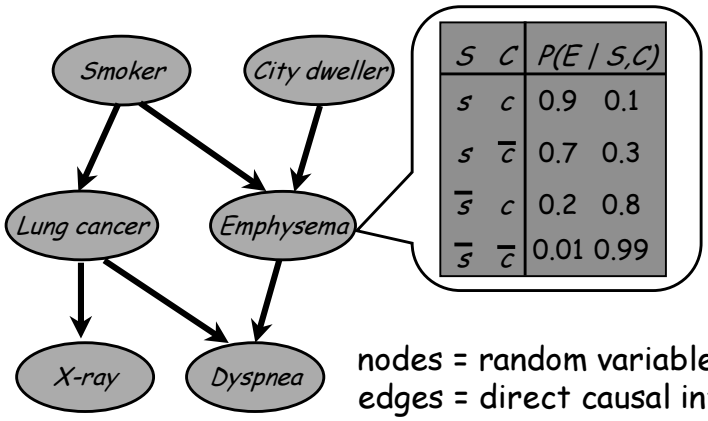
- provide a natural and compact representation of large probability distributions.
- allow for efficient methods for answering a wide range of queries.

Have been used in numerous applications:

- Medical diagnosis
- Hardware diagnosis (Microsoft troubleshooter, NASA Vista project)
- Information retrieval (Ricoh helpdesk)

6

Bayesian networks

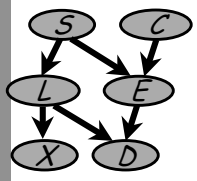


nodes = random variables
edges = direct causal influence

network structure encodes conditional independencies:
X-ray is conditionally independent of *Smoking* given *Lung cancer*

Slide courtesy of D. Koller 7

BN semantics



conditional independencies in BN structure + local probability models = full joint distribution over domain

$$P(s,\bar{c},l,\bar{e},\bar{x},d) = P(s)\,P(\bar{c})\,P(l \mid s)$$
$$P(e \mid s,\bar{c})\,P(\bar{x} \mid l)\,P(d \mid l,\bar{x})$$

- Compact & natural representation:
  - nodes have $\leq k$ parents $\Rightarrow O(2^k n)$ vs. $O(2^n)$ params
  - parameters natural and easy to elicit.

Slide courtesy of D. Koller 8

Learning Scenarios

	Known Structure	Unknown Structure
Complete Data	Easy (closed-form eq.) [SL90, aHugin]	Manageable (discrete search) [B91, CH92, HGC94]
Incomplete Data	Manageable (parametric opt.) [L95, RBKK95]	Hard This Talk

9

Learning Structure

Two approaches:

1. Constraint-based [VP91, SGS93]

- Test for (in)dependencies in the data (e.g., using statistical hypothesis testing).
- Find a structure that is consistent with all dependencies found.

2. Optimization-based [CL68, CH92]

- Define a score to evaluate candidates (w.r.t. data)
- Search for candidate(s) that maximizes score

10

Scores for Bayesian Networks

BIC/MDL score [B94,LB94,S93]:

$$\mathrm{BIC}(G,\Theta:D) = \log P(D \mid G,\Theta) - \frac{\log N}{2} \#\,\text{param in } G$$

Fitness to data

Complexity regularization

Bayesian score [CH92,HGC94]:

$$\mathrm{BDe}(G:D) = \log \int P(D \mid G,\Theta) P(\Theta \mid G) d\Theta + \log P(G)$$

11

Complete Data

Both scores decompose:

- The score can be written a sum of terms, each evaluating the fitness of a node given its parents.

$$\mathrm{Score}(B:D) = \sum_X \mathrm{Score}(X \mid \mathrm{Pa}(X) : N(X,\mathrm{Pa}(X)))$$

Where $N(X,\mathrm{Pa}(X))$ are sufficient statistics for evaluating the family $P(X|\mathrm{Pa}(X))$.

For discrete multinomial BNs, these are counts:

$D =$

X	Y	Z
H	T	T
T	T	T
H	H	H
H	T	T
T	T	H

$N(X)$

X	#
H	3
T	2

$N(X,Y)$

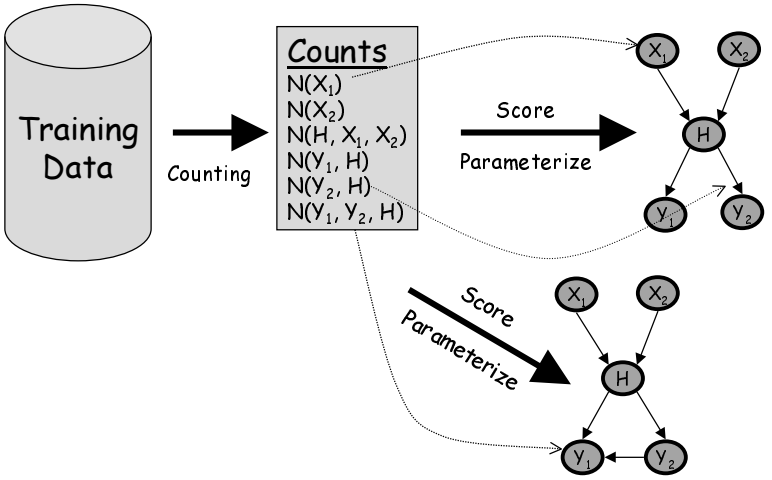
X	Y	#
H	H	1
T	H	0
H	T	2
T	T	2

$N(X,Z)$

X	Z	#
H	H	1
T	H	0
H	T	2
T	T	2

12

Learning from Complete Data



13

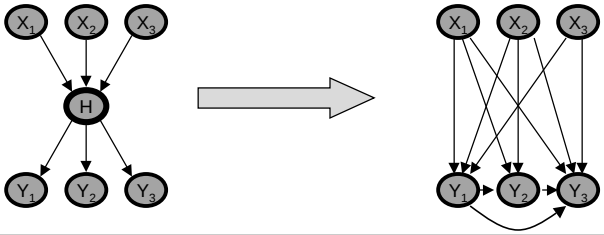
Incomplete Data

Training data is often incomplete

Missing Values

- Medical records
- Survey data

Hidden (latent) Variables



14

Learning From Incomplete Data

Few conceptual difficulties.

- We use score functions based on the likelihood of the observed variables.
- For example, the BIC/MDL score.

$$\mathrm{BIC}(\mathcal{B} : D) = \log P(D \mid \mathcal{B}) - \frac{\log N}{2} \#\,\text{param in}\ \mathcal{B}$$

This ensures that the learned network models the distribution over observed variables.

Assumption: data *missing at random* (MAR) [Rubin]

- Always true for hidden variables.
- Sometimes true for missing values.

15

Unfortunately, incomplete data introduces computational problems.

The likelihood of the data does not decompose:

Complete data:

$$\log P_{\mathcal{B}}(x_1, \ldots, x_n) = \sum_i \log P(x_i \mid \mathrm{Pa}(x_i))$$

Incomplete data:

$$\log P_{\mathcal{B}}(x_1, \ldots, x_k) = \log \sum_{x_{k+1}, \ldots, x_n} \prod_i P(x_i \mid \mathrm{Pa}(x_i))$$

16

Learning Structure From Incomplete Data

Current Approaches: Perform search over possible network structures.

For each candidate structure G:

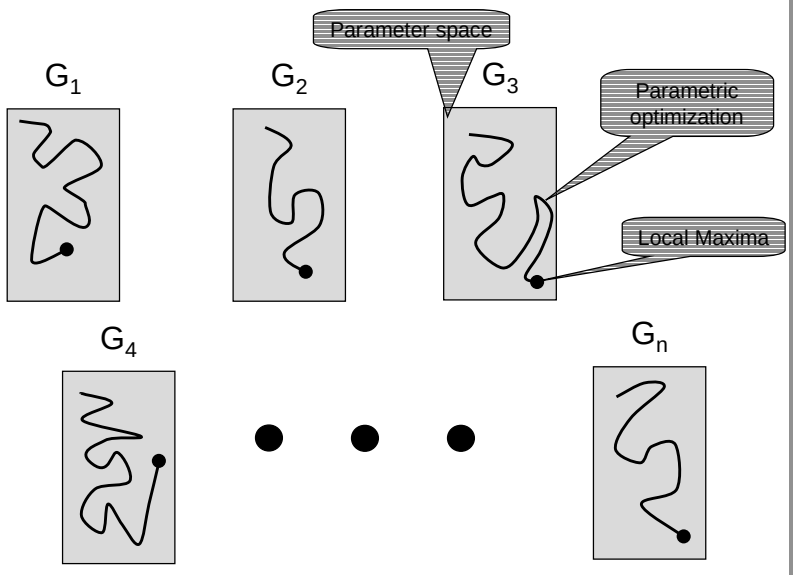
- Find optimal parameters Θ^* for G

  Using non-linear optimization methods:

  - Gradient Ascent [RBKK95]
  - Expectation Maximization (EM) [DLR77, L95]
- Score G using these parameters.

  Body of work on efficient approximations for the score, in particular for the Bayesian score. [HGM96,CH96]

17



18

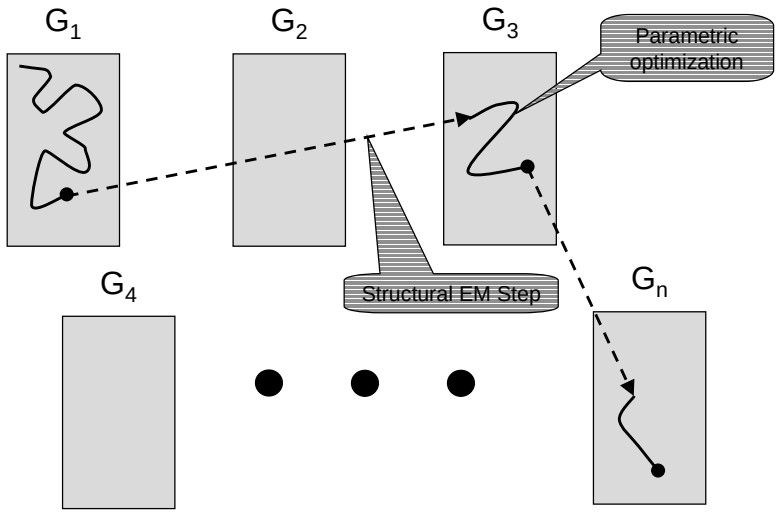
Problem

Such procedures are computationally expansive!

- Computation of optimal parameters, per candidate, requires non-trivial optimization step.
- Spend non-negligible computation on a candidate, even if it is a low scoring one.

In practice, such learning procedures are feasible only when we consider small sets of candidate structures (e.g., Autoclass).

19



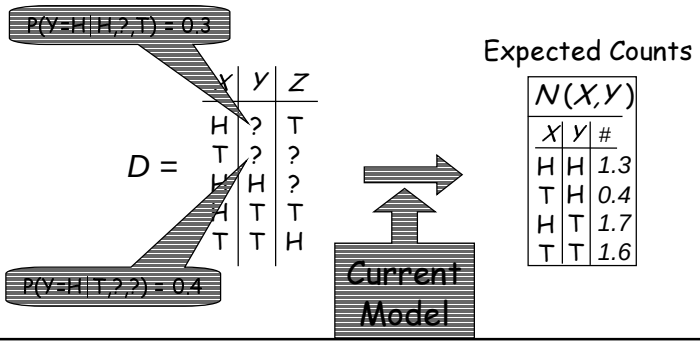
20

Expectation Maximization

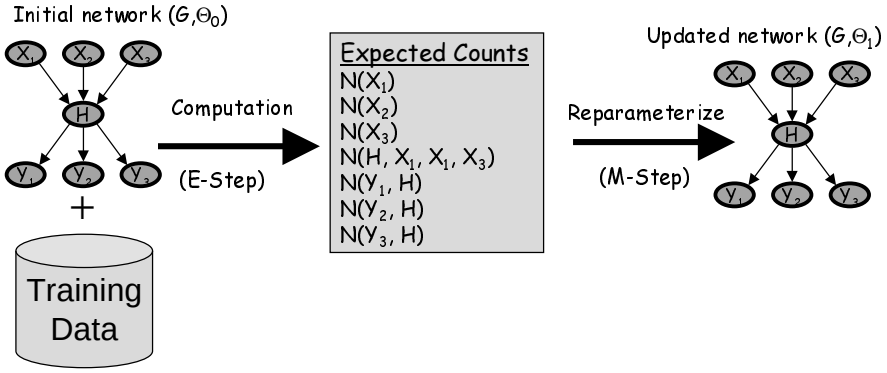
A general method for optimizing parameters.

Use an initial guess for parameters to find better ones.

Rough idea: use current parameters to "complete" counts.



EM (continued)



Structural EM

Idea: Use parameters found for previous structures to help evaluate new structures.

Scope: searching over structures over the same set of random variables.

Outline:

- Perform search in (Structure, Parameters) space.
- Use EM-like iterations, using previously best found solution as a basis for finding either:
  - Better scoring parameters --- "parametric" EM step

  or
  - Better scoring structure --- "structural" EM step

Structural EM Step

Assume $B_0 = (G_0, \Theta_0)$ is "current" hypothesis.

Goal: Maximize expected score, given B_0

$$E[\text{Score}(B : D^+) \mid D, B_0] = \sum_{D^+} \text{Score}(B : D^+) P(D^+ \mid D, B_0)$$

where D^+ denotes completed data sets.

Theorem:(progress)

If $E[\text{Score}(B : D^+) \mid D, B_0] > E[\text{Score}(B_0 : D^+) \mid D, B_0]$

$\Rightarrow \text{Score}(B : D) > \text{Score}(B_0 : D)$.

- This implies that by improving the expected score, we find networks that have higher objective score.

Corollary: repeated applications of this EM step will converge.

Structural EM for BIC/MDL

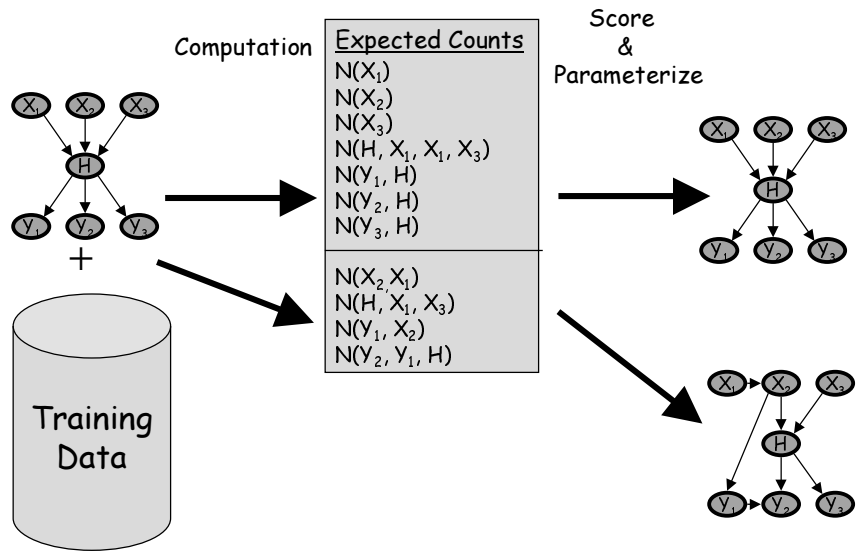
For the BIC/MDL score, we get that

$$
\begin{aligned}
&E[\mathrm{BIC}(B : D^{+}) \mid D, B_0] \\
&\quad = E[\log P(D^{+} \mid B) \mid D, B_0] - \mathrm{Penalty}(B) \\
&\quad = E\Big[\sum_i N(X_i, Pa_i) \log P(X_i \mid Pa_i) \mid D, B_0\Big] - \mathrm{Penalty}(B) \\
&\quad = \sum_i E[N(X_i, Pa_i) \mid D, B_0] \log P(X_i \mid Pa_i) - \mathrm{Penalty}(B)
\end{aligned}
$$

Consequence:

- We can use complete-data methods, were we use expected counts, instead of actual counts.

25



26

Structural EM in Practice

- E-Step: compute expected counts for all candidate structures.
- M-Step: choose structure that maximizes expected score.

Problem: there are (exponentially) many structures.

- We cannot computed expected counts for all of them in advance.

Solution:

- M-Step: search over network structures (e.g., hillclimbing).
- E-Step: on-demand, for each structure G examined by M-Step, compute expected counts.
- Use smart caching/prefetching schemes to minimize overall computations.

27

The Structural EM Procedure

Input: $B_0 = (G_0,\Theta_0)$
loop for $n = 0,1,\ldots$ until convergence
Improve parameters:
$\Theta'_n = $ Parametric-EM (G_n,Θ_n)
let $B'_n = (G_n,\Theta'_n)$
Improve structure:
Search for a network $B_{n+1} = (G_{n+1},\Theta_{n+1})$ s.t.
$E[Score(B_{n+1}:D) \mid B'_n] > E[Score(B'_n:D) \mid B'_n]$

- Parametric-EM() can be replaced by Gradient Ascent, Newton-Raphson methods, or accelerated EM.
- Early stopping parameter optimization stage avoids "entrenchment" in current structure.

28

Convergence Properties

Theorem: The SEM procedure converges in score:

The limit $\lim_{n\to\infty} \text{Score}(B_n : D)$ exists.

Theorem: Convergence point is a local maxima:

If $G_n = G$ infinitely often, then, Θ_G, the limit of the parameters in the subsequence with structure G, is a stationary point in the parameter space of G.

SEM for the Bayesian Score

Theorem: If $E[BDe(G:D^+)|D,G_0] > E[BDe(G_0:D^+)|D,G_0]$

$\Rightarrow BDe(G : D) > BDe(G_0 : D)$.

Where $E[\bullet|D, G_0]$ is expectation based on

$$P(\bullet \mid D, G_0) = \int P(\bullet \mid G_0, \Theta) P(\Theta \mid D, G_0) d\Theta$$

Expanding the expected score:

$$\begin{aligned}
&E[BDe(G : D^+) \mid D, G_0] \\
&\quad = E[\log \int P(D^+ \mid G, \Theta) P(\Theta \mid G) d\Theta \mid D, G_0] + \log P(G) \\
&\quad = E[\log \prod_i \prod_{\text{pa}_i} \frac{\Gamma(\alpha_{\text{pa}_i})}{\Gamma(\alpha_{\text{pa}_i} + N(\text{pa}_i))} \prod_{x_i} \frac{\Gamma(\alpha_{x_i,\text{pa}_i} + N(x_i, \text{pa}_i))}{\Gamma(\alpha_{x_i,\text{pa}_i})}] + \log P(G) \\
&\quad = \sum_i \sum_{\text{pa}_i} (\sum_{x_i} E[\log \Gamma(\alpha_{x_i,\text{pa}_i} + N(x_i, \text{pa}_i))] - E[\log \Gamma(\alpha_{\text{pa}_i} + N(\text{pa}_i))]) + \log P(G)
\end{aligned}$$

Two problems:

- How do we compute $P(\bullet | D, G_0)$?

  Approximation: use MAP parameters

  $P(\bullet | D, G_0) \approx P(\bullet | G_0, \Theta^*)$, where Θ^* maximizes $P(\Theta | D, G_0)$.

  We can find Θ^* using standard EM.

- How do we compute $E[\log\Gamma(\alpha+N(x))]$?

  Approximation: $N(x)$ is the sum of independent indicator variables: $N(x) = \Sigma_j I_j(x)$

  By Central Limit Theorem

  $$N(x) \sim \text{Normal}(\Sigma_j E[I_j(x)], \Sigma_j \text{Var}[I_j(x)])$$

  Use numerical integration (e.g., Gauss-Hermite quadrature) to compute $E[\log\Gamma(\alpha+N(x))]$, under this assumption.

31

Using Structural EM

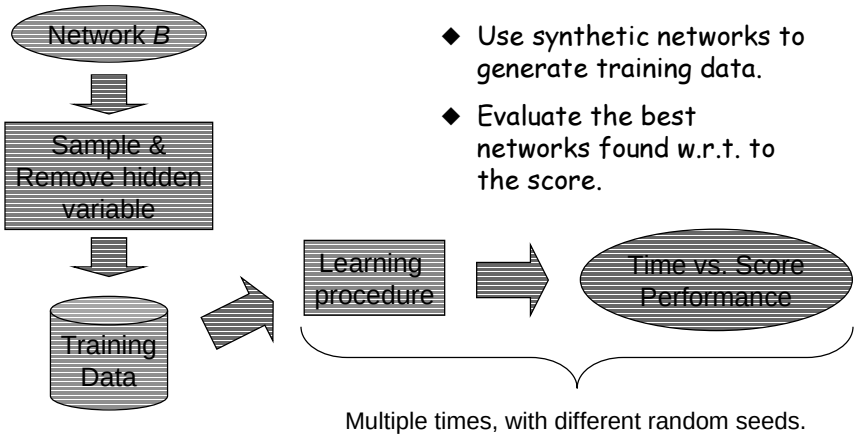
Problem: Structural EM is susceptible to run into local maxima.

Solution: Combine Structural EM with

- Random perturbations:
  - At convergence point, randomly add/reverse arcs in neighborhood of hidden variables.
  - Then, restart Structural EM from the perturbed structure.
- Random restarts:
  - Perform random walk (adding/removing/reversing arcs) from best structure so far.
  - Then, restart uphill search.

32

Experimental Setup



- Use synthetic networks to generate training data.
- Evaluate the best networks found w.r.t. to the score.

Experiments---Methods

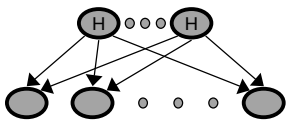
Iterated-EM ("standard" approach)

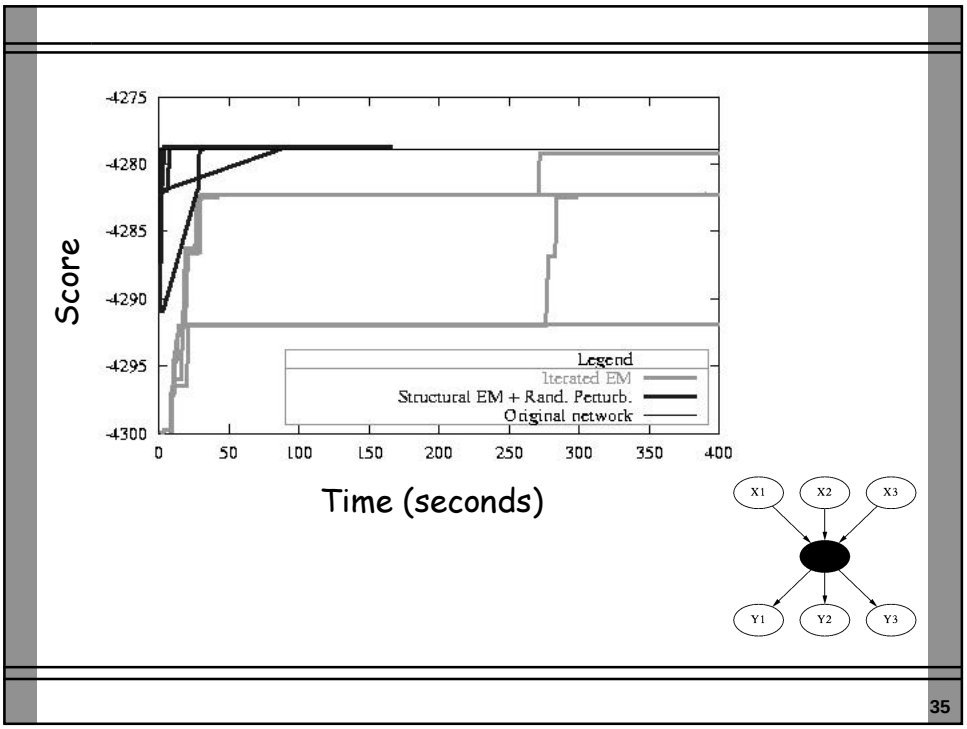
- Evaluate each structure using parametric EM
- Greedy hillclimbing search over structures with random restarts at local maxima

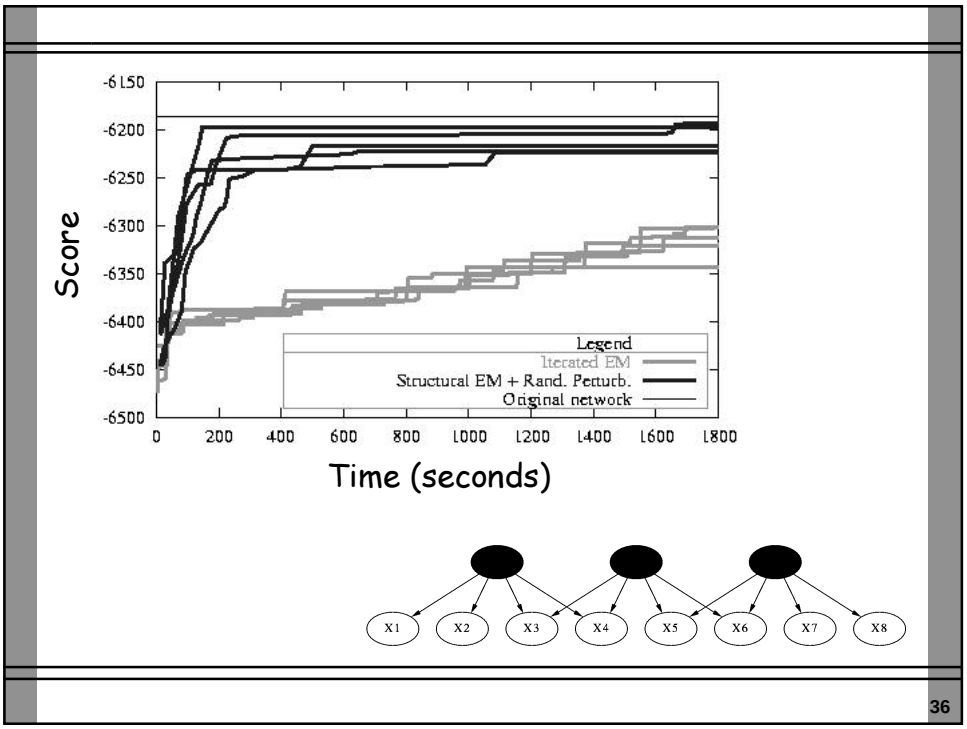
Structural-EM + Perturb

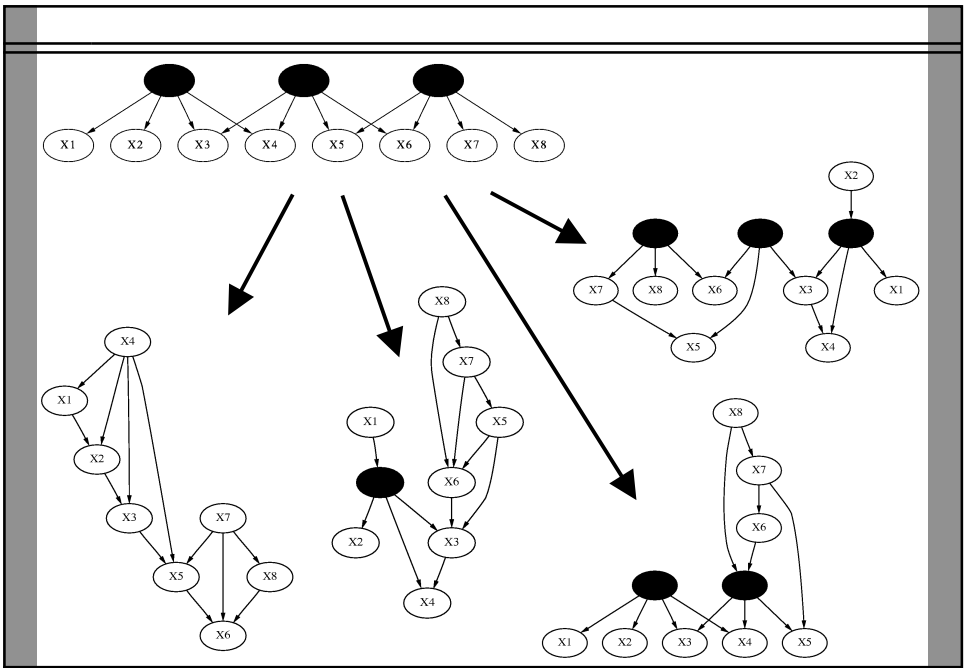
- Apply Structural EM to learn structures.
- Applies random perturbations and random restarts.

Initial structure:

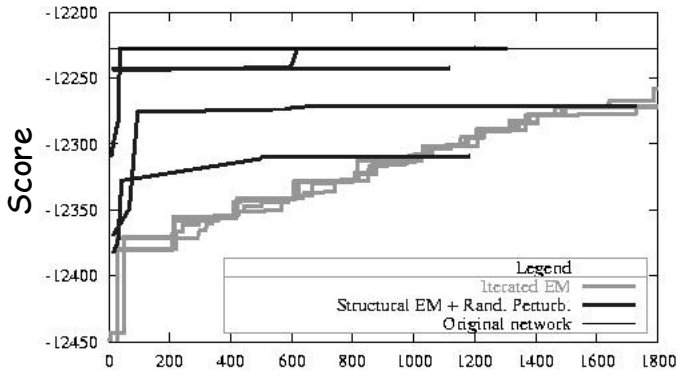


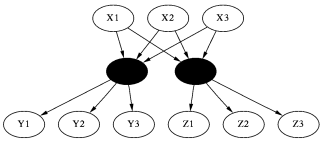

Score
Time (seconds)
-4275
-4280
-4285
-4290
-4295
-4300
0
50
100
150
200
250
300
350
400
Legend
Iterated EM
Structural EM + Rand. Perturb.
Original network
X1
X2
X3
Y1
Y2
Y3
35


Score
Time (seconds)
-6150
-6200
-6250
-6300
-6350
-6400
-6450
-6500
0
200
400
600
800
1000
1200
1400
1600
1800
Legend
Iterated EM
Structural EM + Rand. Perturb.
Original network
X1
X2
X3
X4
X5
X6
X7
X8
36

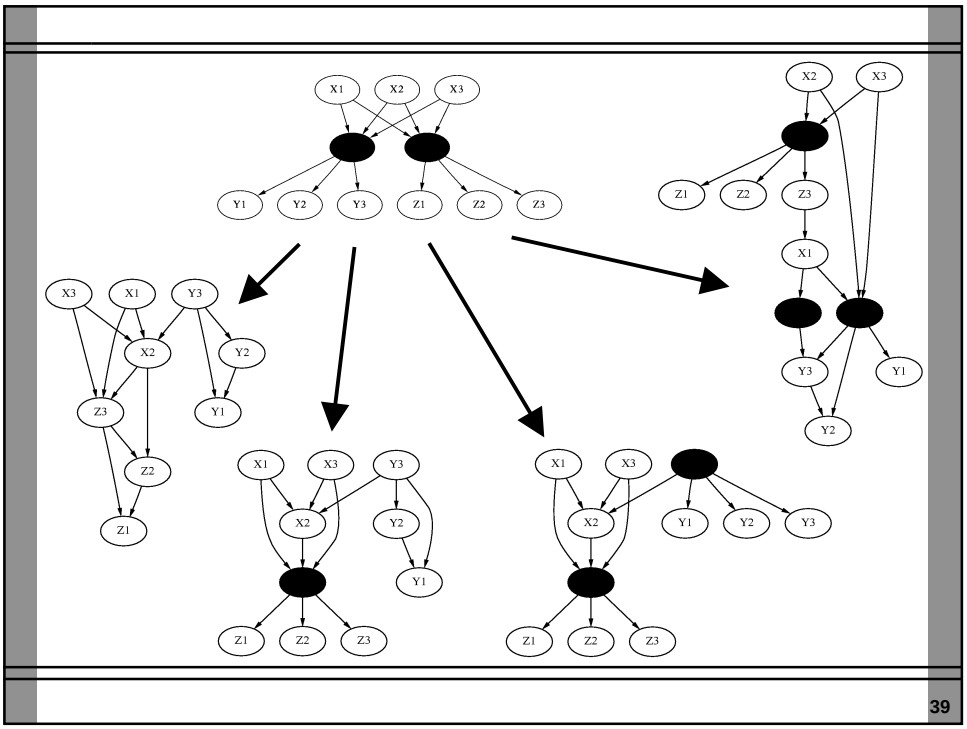

X1 X2 X3 X4 X5 X6 X7 X8
X2 X7 X8 X6 X3 X1 X5 X4
X4 X1 X2 X3 X7 X5 X8 X6
X8 X7 X1 X5 X6 X2 X3 X4
X8 X7 X6 X1 X2 X3 X4 X5

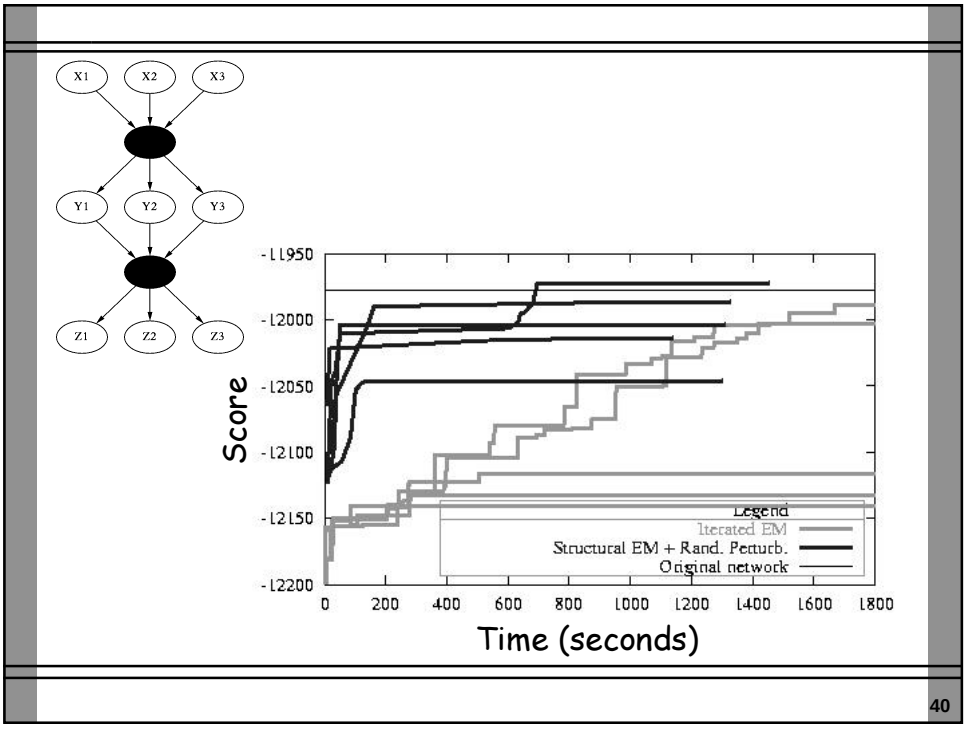
37

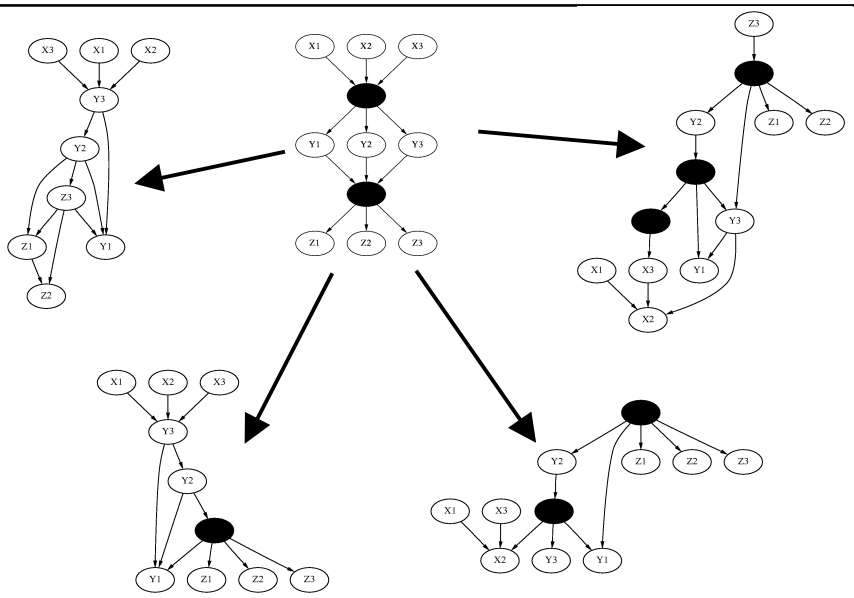

-12200
-12250
-12300
-12350
-12400
-12450
Score
0 200 400 600 800 1000 1200 1400 1600 1800
Time (seconds)
Legend
Iterated EM
Structural EM + Rand. Perturb.
Original network


X1 X2 X3
Y1 Y2 Y3 Z1 Z2 Z3

38


X1
X2
X3
Y1
Y2
Y3
Z1
Z2
Z3
39


X1
X2
X3
Y1
Y2
Y3
Z1
Z2
Z3
-11950
-12000
-12050
-12100
-12150
-12200
Score
0
200
400
600
800
1000
1200
1400
1600
1800
Time (seconds)
Legend
Iterated EM
Structural EM + Rand. Perturb.
Original network
40



41

Summary

Structural EM:

- provides significant computational advantages over existing approaches
- allows to quickly "adjust" a network to combine a fixed number of hidden variables

Experimental results show effectiveness in learning from non-trivial domains.

42

Discussion & Future Work

Work in progress:

- Better suited exact and approximate inference for the "structural" EM step.
- "smarter" local search procedures.
- Large scale experiments.
- *Dynamic Probabilistic Networks* and *Phylogenetic Trees*.

For effective learning of hidden structure, we also need:

- Prior knowledge (whenever possible).
- Heuristic guidance based on (in)dependence properties observed in the data (e.g., using methods of [SGS93]).
- Ways of introducing additional hidden variables "on the fly" during learning.

43