Graphical Models and Bayesian Networks

Bioinformatics
Course
Supplement

Outline

- Introduction
- Graphical Models
- Bayesian Networks
- EM Algorithms
- Summary

Learning and Probabilistic Inference

- Many probabilistic inference problems can be solved if we have a joint distribution of the variables involved.
- Joint distribution associated with a probabilistic inference problem can be decomposed into locally interacting factors.
- By taking advantage of probabilistic structure, inference can be performed more efficiently than the blind.application of Bayes' rule.
- Many learning problems can be formulated in the framework of probabilistic inference:
  - Supervised learning (classification)
  - Unsupervised learning
  - Data compression
  - Channel coding

Supervised Learning (Classification)

- A classifier
  - Estimates the probability $P_r(j \mid \mathbf{v})$
  - Bayes optimal classifier
    - A minimum error rate is achieved by choosing the class j that maximizes $P_r(j \mid \mathbf{v})$

j: class index
v: visible variables
h: hidden variables

- Example: digit classification
  - $P(j)$: a priori distribution over the digit classes
  - $P(\mathbf{h} \mid j)$: a distribution over a set of hidden attributes of a digit class
  - $P(\mathbf{v} \mid \mathbf{h})$: a distribution over possible images given a set of features

$$P(j, \mathbf{h}, \mathbf{v}) = P(j)P(\mathbf{h} \mid j)P(\mathbf{v} \mid \mathbf{h})$$

$$P(j \mid \mathbf{v}) = \frac{P(j, \mathbf{v})}{\sum_{j'} P(j', \mathbf{v})} = \frac{\sum_{\mathbf{h}} P(j, \mathbf{h}, \mathbf{v})}{\sum_{j'} \sum_{\mathbf{h}} P(j', \mathbf{h}, \mathbf{v})}$$

Unsupervised Learning

- The Goal
  - Process a set of training data and then extract underlying structure which is believed to be relevant to perception and conception.
- A Possible Approach
  - In addition to the input sensory variables $\mathbf{v}$ there are hidden "concept" variables $\mathbf{h}$
  - $\mathbf{v}, \mathbf{h}$ are linked together by a parameterized probability model $P(\mathbf{v}, \mathbf{h} \mid \theta)$
  - Given the hidden variables, the sensory inputs are independent.

$$P(\mathbf{v}, \mathbf{h}, \theta) = P(\mathbf{h} \mid \theta) \prod_{i=1}^{N} P(v_i \mid \mathbf{h}, \theta)$$

  - Maximum likelihood parameter estimation

$$\theta^{ML} = \arg\max \prod_{t=1}^{T} P(\mathbf{v}(t) \mid \theta) \qquad P(\mathbf{v}(t) \mid \theta) = \sum_{\mathbf{h}} P(\mathbf{v}(t), \mathbf{h} \mid \theta)$$

Probabilistic Sturucture and Graphical Models

- Probabilistic structure
  - Can be characterized by a set of conditional independence.
  - Express the joint distribution as a product of factors.
  - Each factor depends on a subset of the random variables.
  - Takes advantage of a graphical description of the dependencies between random variables.
- Graphical models
  - Graph theory provides a succinct way to represent probabilistic structure.
  - A graphical representation for probabilistic structure + functions that can be used to derive the joint distribution.
  - Concisely capture probabilistic structure, and forms a framework for computing useful probabilities.

Graphical Models

Graphical Models

- Graphical models are a marriage between graph theory and probability theory
- They clarify the relationship between neural networks and related network-based models such as HMMs, MRFs, and Kalman filters
- Indeed, they can be used to give a fully probabilistic interpretation to many neural network architectures
- Some advantages of the graphical model point of view
  - Inference and learning are treated together
  - Supervised and unsupervised learning are merged seamlessly
  - Missing data handled nicely
  - A focus on conditional independence and computational issues
  - Interpretability (if desired)

Learning and Inference in Graphical Models

- A key insight from the graphical model point of view:

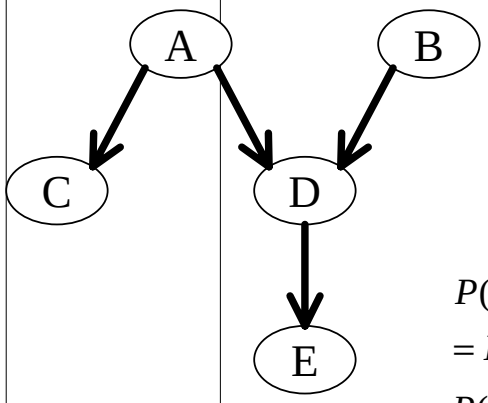
  It is not necessary to learn that which can be inferred

- The weights in a network make local assertions about the relationships between neighboring nodes
- Inference algorithms turn these local assertions into global assertions about the relationships between nodes
  - e.g., correlations between hidden units conditional on an input-output pair
  - e.g., the probability of an input vector given an output vector
- This is achieved by associating a joint probability distribution with the network

Graphical Models and Independence

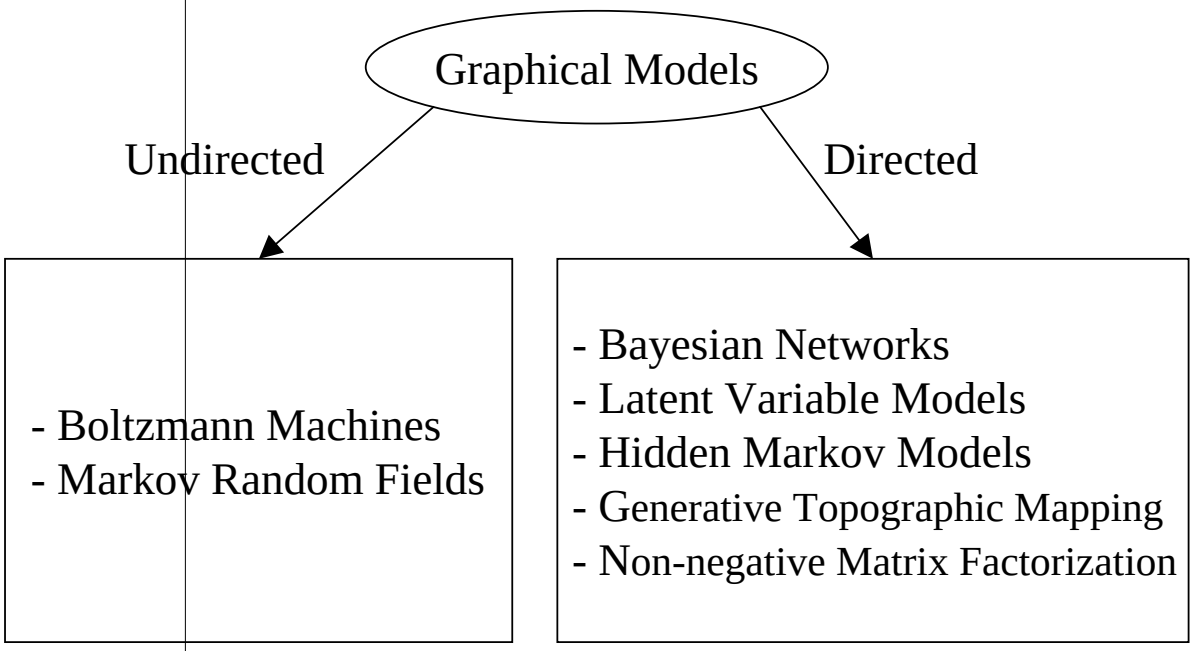
- The most common simplifying trick
  - Some independence between the variables
  - Some conditional independence of subsets of variables, conditioned on other subsets of variables
- Graph
  - Node: variable
  - Missing edge: independence relationship
  - Independent relationship: the global high-dimensional probability distribution *P* over all variables can be *factored* into a product of *simpler local probability distributions* over lower-dimensional spaces associated with smaller clusters of variables

Graphical Model Structure: An Example



$$P(A,B,C,D,E)$$
$$= P(A)P(B \mid A)P(C \mid A,B)P(D \mid A,B,C)P(E \mid A,B,C,D)$$
$$P(A,B,C,D,E)$$
$$= P(A)P(B)P(C \mid A)P(D \mid A,B)P(E \mid D)$$
$$P(\mathbf{X}) = \prod_{i=1}^{n} P(X_i \mid \mathbf{Pa}_i)$$

Classes of Graphical Models



Graphical Models: Undirected and Directed

- Undirected models
  - Edge: symmetric interactions
  - Statistical mechanics, image processing
  - Markov random fields, undirected probabilistic independence networks, Boltzmann machines, Markov networks, log-linear models
- Directed models
  - Edge: not symmetric interactions, causal relationship, time irreversibility
  - Expert systems, problems based on temporal data
  - Bayesian networks, belief networks, directed probabilistic independence networks, causal networks, influence diagrams

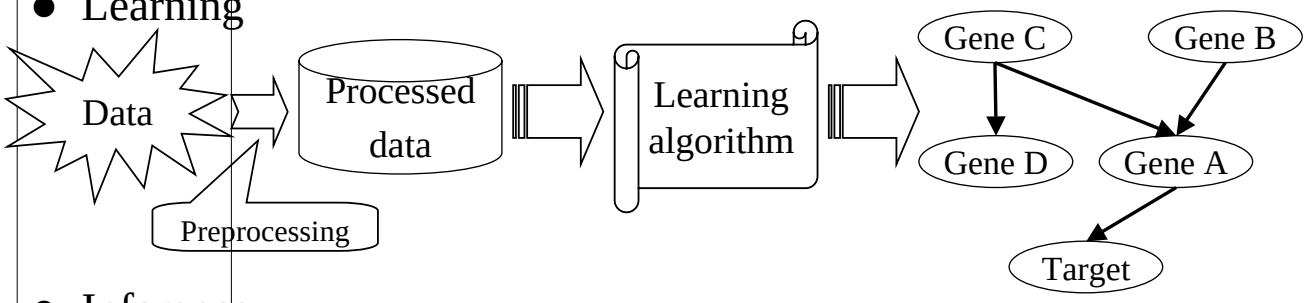
Applications of Graphical Models

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

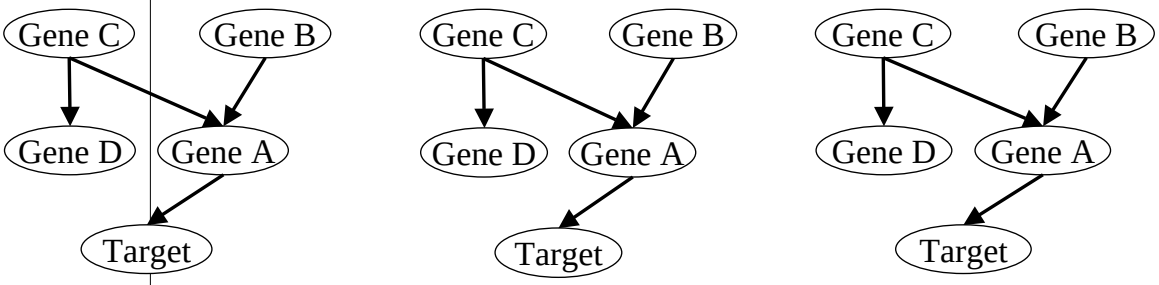
Graphical Models for Gene Expression Analysis

[Hwang et al., 2001]

- Learning



- Inference

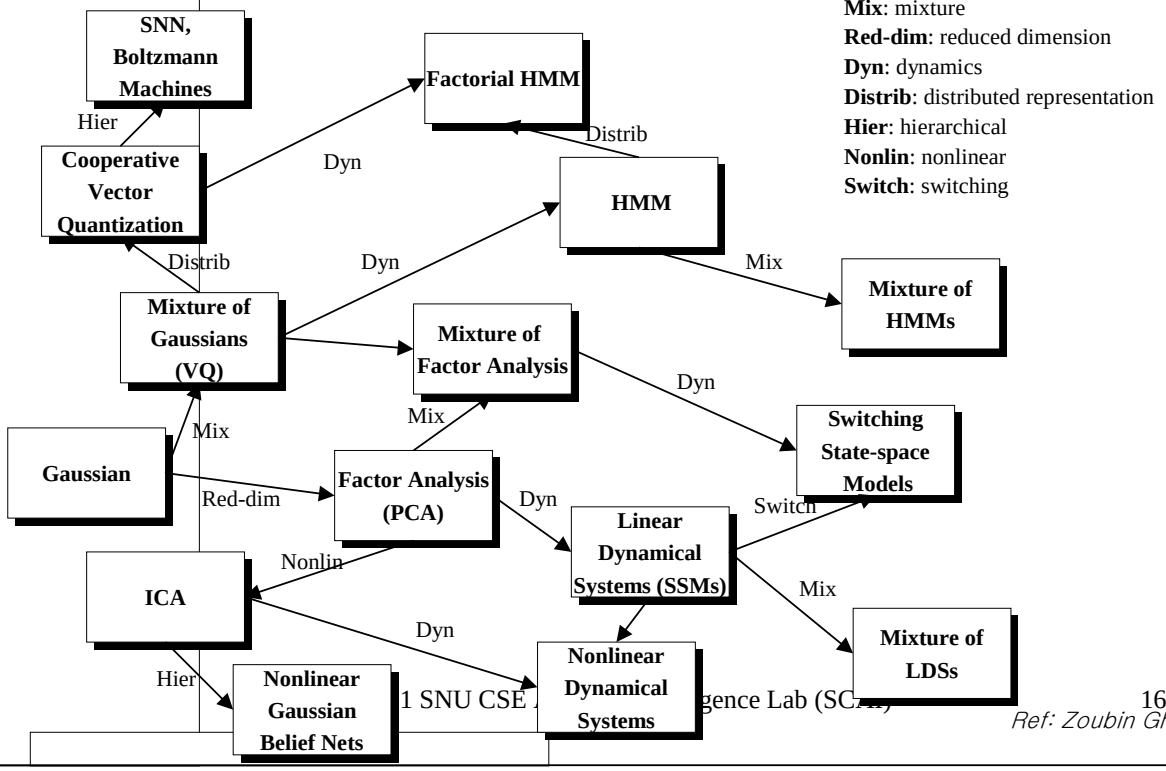


The values of Gene C and Gene B are given.

Belief propagation

Probability for the target is computed.

Variations of (Generative) Graphical Models



Mix: mixture
Red-dim: reduced dimension
Dyn: dynamics
Distrib: distributed representation
Hier: hierarchical
Nonlin: nonlinear
Switch: switching

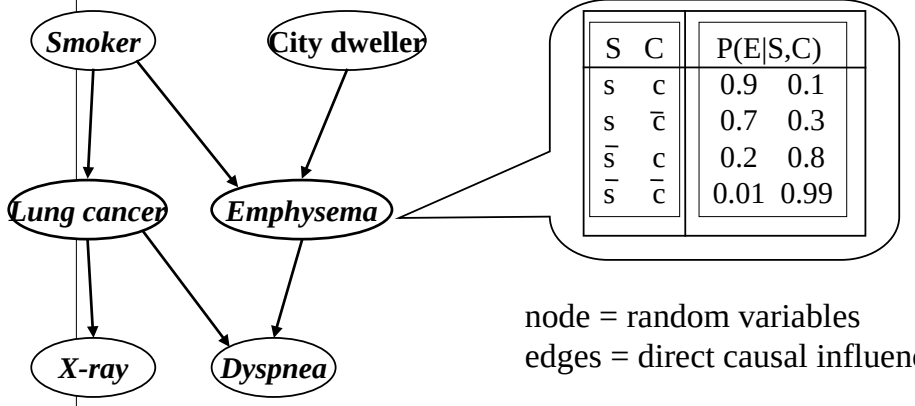
Ref: Zoubin Ghahramani

Bayesian Networks

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 method for answering a wide range of queries.
- Have been used in numerous applications:
  - Medical diagnosis (PathFinder, QMR)
  - Hardware diagnosis (Microsoft troubleshooter, NASA/Rockwell Vista project)
  - Information retrieval (Ricoh helpdesk)

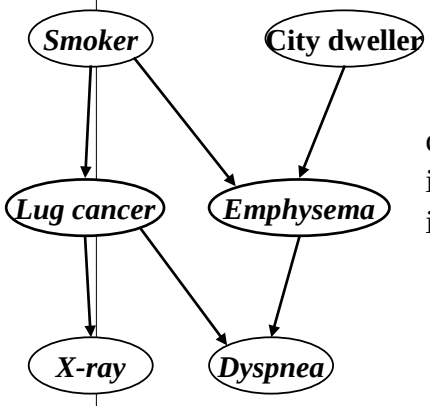
Bayesian Networks: Example



node = random variables
edges = direct causal influence

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

Bayesian Networks: Semantics



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

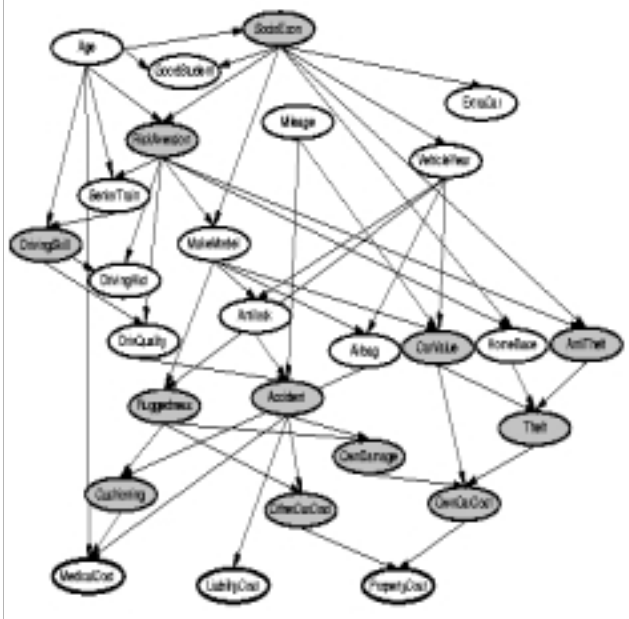
$$P(s,\bar{c}\mid\bar{e},\bar{x},d)=P(s)P(\bar{c})P(l\mid s)$$
$$P(e\mid s,\bar{c})P(\bar{x}\mid l)P(d\mid l,\bar{e})$$

- Compact & natural representation:
  - Nodes have $\leq k$ parents => $O(2^k n)$ vs. $O(2^n)$ parents
  - Parameters natural and easy to elicit

Bayesian Networks: 3 Main Tasks for any Bayes Net Toolbox

- Model specification
  - Qualitative: graph structure (DAG)
  - Quantitative: parameters of the conditional probability distributions (CPDs), i.e., Pr(node|parents)
- Inference
  - Goal: compute $P(Q|V) = \Sigma_h P(h, Q|V)$
  - Different algorithms make different tradeoffs between simplicity, generality, accuracy, speed, etc.
- Learning
  - Qualitative: graph structure (hard)
  - Quantitative: parameters of the CPDs (easy)
  - Learning with partial observability uses inference as a subroutine

Bayesian Networks: Model Specification I - Structure



- The structure is a directed acyclic graph (DAG), represented as a (sparse) adjacency matrix
- The nodes must always be numbered in topological order, i.e., ancestors before descendants
  - Future work:
  - Add a GUI
  - Add a file parser for some of the standard formats

Bayesian Networks: Model Specification I - Structure

- Tabular (multinomial)
- Gaussian
- Softmax (logistic/sigmoid)
- Noisy-or
- Deterministic
- Fully object-oriented, so easy to new CPDs

Bayesian Networks: Inference I

- Exact inference for static BNs
  - Junction tree
  - Variable elimination
  - Brute force enumeration (for discrete nets)
  - Linear algebra (for Gaussian nets)
  - Pear‘s algorithm (for polytrees)
  - Quickscore (for QMR)
- Approximate inference for static BNs
  - Likelihood weighting
  - Loopy belief propagation
- Each inference engine is an object, so easy to add more.

Bayesian Networks: Inference II

- Each inference algorithm is implemented in BNT as an "inference engine" object in the following respects [Gatsby Neuroscience Unit, UCL].
- Designed for static or dynamic models?
- Exact or approximate inference?
- Works for all topologies or makes restrictions?
- Works for all node types or makes restrictions?
- Handles any pattern of evidence, or must be fixed?
- What computation is done when the following are specified
  - Structure
  - Parameters
  - Evidence
  - Query

Bayesian Networks: Inference III

Name	Exact?	Node type?	Topology
cond_gauss	exact	CG	all
enumerative	exact	allD	all
gaussian	exact	allG	all
jtree	exact	D,G,CG	all
var_elim	exact	D,G,CG	all
pearl	exact	D,G	polytree
quickscore	exact	D,G	QMR
lik_weight	approx	any	all
loopy_pearl	approx	D,G	all

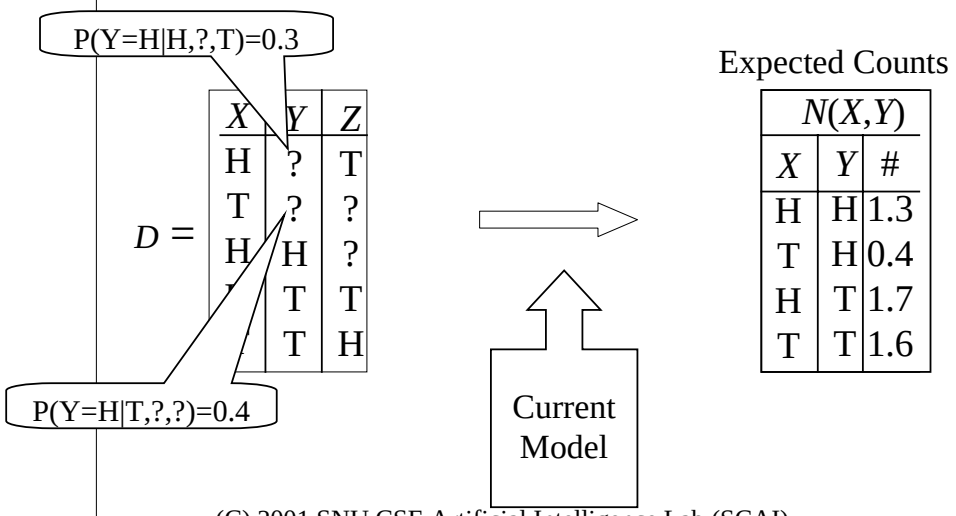
(Zoubin Ghahramani 2000)

Bayesian Networks: Learning

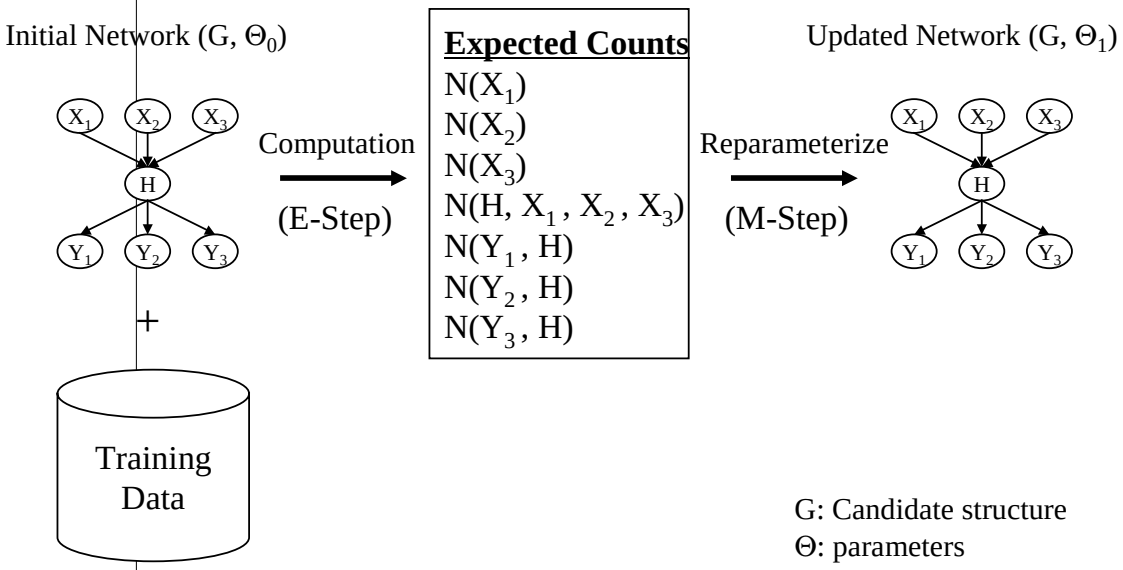
- Algorithms
  - Batch EM parameter learning
  - Sequential Bayesian parameter learning
  - Structure learning (for observed tabular nodes only)
- Regularization
  - Any node can have its parameters clamped (made non-adjustable)
  - Any set of compatible nodes can have their parameters tied (cf. weight sharing in a neural net)
  - Some node types (e.g., tabular) supports priors for MAP estimation
  - Gaussian covariance matrices can be declared full or diagonal, and can be tired across states of their discrete parents (if any)
- Modularity
  - Each node type has its own M method, e.g. softmax nodes use IRLS
  - Each inference engine implements its own E method

Expectation Maximization (EM)

- 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 (Cont'd)

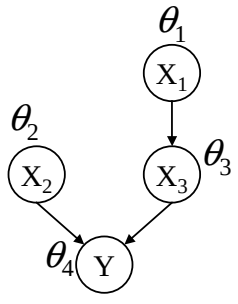


The EM Algorithm for Learning with Hidden Variables:

- Assume a model parameterised by θ with observable variables Y and hidden variables X
- Goal: maximize log likelihood of observables

$$L(\theta) = \ln P(Y \mid \theta) = \ln \sum_X P(Y, X \mid \theta)$$

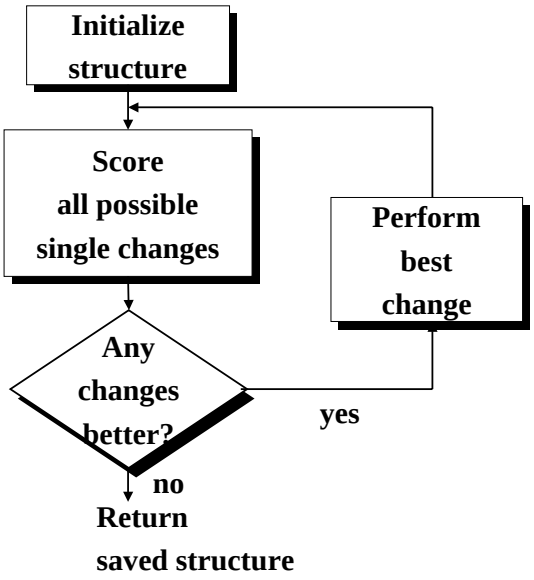
  - E-step: first infer $P(X|Y, \theta_{old})$, then
  - M-step: find θ_{new} using complete data learning



- The E-step requires solving the inference problem: finding explanations, X, for the data, Y given the current model θ

Model Search: Structural Learning

- Finding the BN structure with the highest score among those structures with at most *k* parents is NP hard for *k* >1 (Chickering, 1995)
- Heuristic methods
  - Greedy
  - Greedy with restarts
  - MCMC methods



Bayesian Networks: Current Issues

- Fancy distributions (decision trees, logit, probit, noisy or continuous variables)
- Model selection
- Hidden variables and missing data
- Time-varying domains (cf. dynamic models)
- Online learning
- Causality and causal interpretations for Bayes nets

Summary

- Graphical models provide a principled and rigorous approach to inference and learning.
- They offer a general framework for treating supervised and unsupervised learning together.
- Graphical models are interpretable (cf. multilayer perceptrons).
- Directed graphical models allow for causal relationships to be represented and discovered.
- Graphical models can be used to clarify the relationship between different machine learning methods.
- Drawbacks: computationally intensive
- But, work is in progress to develop efficient learning and inference algorithms for graphical models.