APPROXIMATE INFERENCE VIA VARIATIONAL TECHNIQUES

Michael I. Jordan

Center for Biological and Computational Learning
Massachusetts Institute of Technology
http://www.ai.mit.edu/projects/jordan.html

Acknowledgments:

Zoubin Ghahramani, Tommi Jaakkola, Marina Meila
Lawrence Saul

August, 1997

Introduction

- There are a number of examples of graphical models in which exact inference is efficient:
  - chain-like graphs
  - tree-like graphs
- However, there are also examples of graphical models in which exact inference can be inefficient:
  - dense graphs
  - layered graphs
  - coupled graphs
- A variety of methods are available for *approximate* inference in such situations
- Stochastic methods
  - importance sampling
  - Markov chain Monte Carlo
- Deterministic methods
  - search-based methods
  - bounded conditioning
  - localized partial evaluation
  - variational methods
- Hybrids

Inferential complexity

- Computational complexity of exact computation
  - passing messages requires marginalizing and scaling the clique potentials
  - thus the time required is exponential in the number of variables in the largest clique
- Examples in which the largest clique can be unrealistically large
  - neural networks
  - factorial hidden Markov models
  - hidden Markov decision trees
  - Markov models with long-range connections
  - codes
  - the QMR database

Complexity results for approximate inference?

- Interesting, but taken too seriously they are red herrings
- It is surely true that there are families of probability distributions in which approximate inference is intractable
- But it is also likely that there are useful families of probability distributions for which good approximate inference can be achieved
  - cf. finite element analysis
  - cf. statistical mechanics

Multilayer neural networks as graphical models

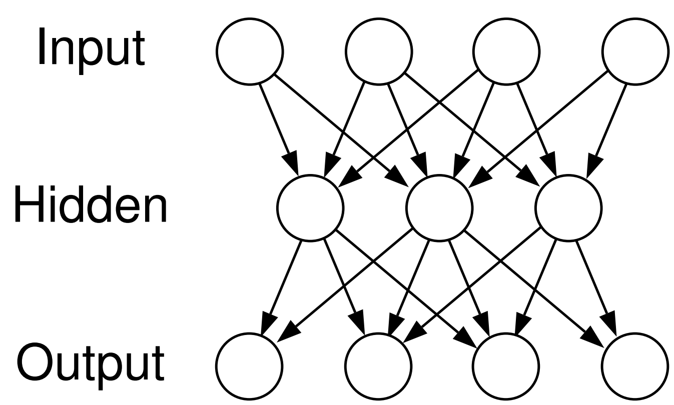
(cf. Neal, 1992; Saul, Jaakkola, & Jordan, 1995)

- Associate with node i a latent binary variable whose conditional probability is given by:

$$P(S_i = 1|\mathbf{x}_i) = \frac{1}{1 + e^{-\theta_i^T \mathbf{x}_i}}$$

where $\mathbf{x}_i$ are the parents of node i

- A multilayer neural network with logistic hidden units:

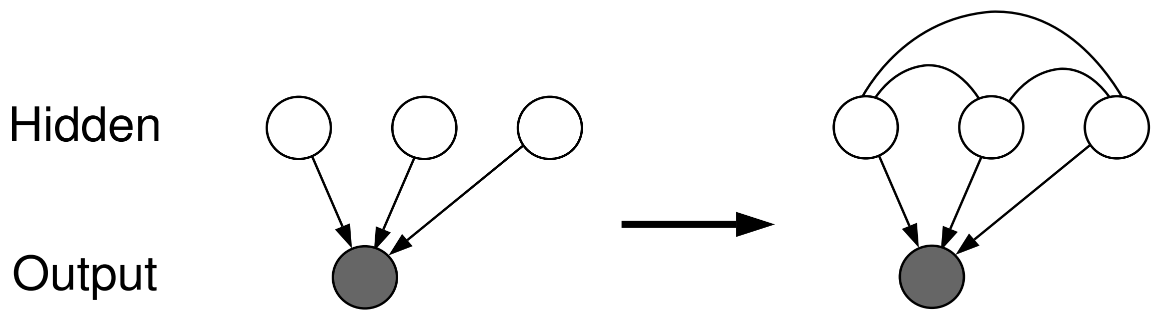


- has a joint distribution that is a product of logistic functions:

$$P(\{S_1, S_2, \ldots, S_N\}) = \prod_i \left[\frac{1}{1 + e^{-\theta_i^T \mathbf{x}_i}}\right]$$

Complexity of neural network inference

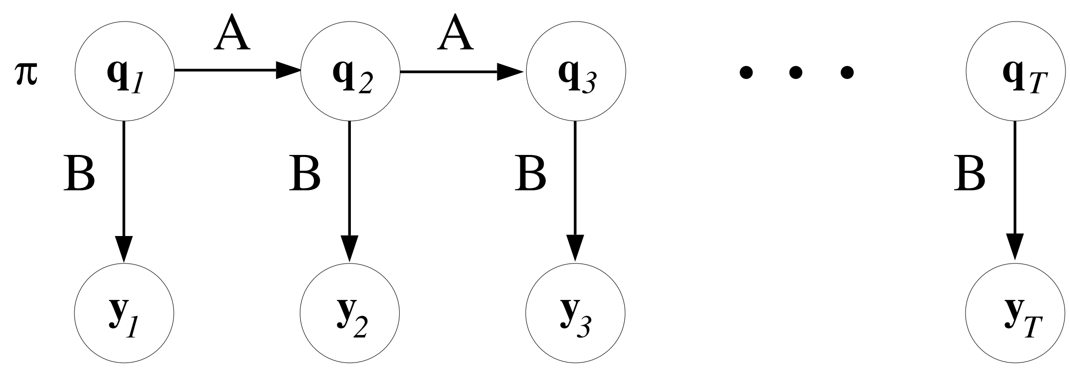
- When an output node is known (as it is during learning), moralization links the hidden units:



- So inference scales at least as badly as $O(2^N)$, where N is the number of hidden units
- And triangulation adds even more links

Example: Hidden Markov models

- A hidden Markov model is a popular time series model
- It is a "mixture model with dynamics"



- T time steps
- M states ($\mathbf{q}_t$ is a multinomial RV)
- N outputs ($\mathbf{y}_t$ is a multinomial RV)
- state transition probability matrix A:

$$A = P(\mathbf{q}_{t+1}|\mathbf{q}_t)$$

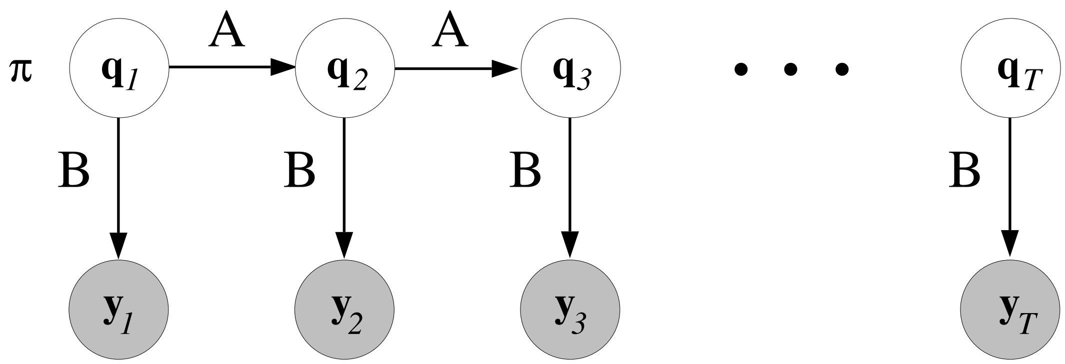
- emission matrix B:

$$B = P(\mathbf{y}_t|\mathbf{q}_t)$$

- initial state probabilities π:

$$\pi = P(\mathbf{q}_1)$$

HMM probability calculations

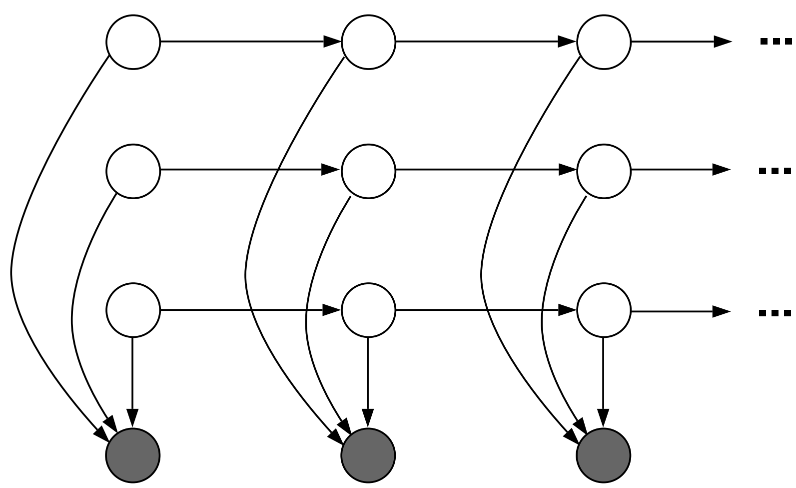


- The time series of $\mathbf{y}_t$ values is the *evidence*
- The *inference* calculation involves calculating the probabilities of the hidden states $\mathbf{q}_t$ given the evidence
- The classical algorithm for doing this calculation is the *forward-backward* algorithm
  - it is a special case of the junction tree algorithm

Factorial hidden Markov models

(Williams & Hinton, 1991; Ghahramani & Jordan, 1996)

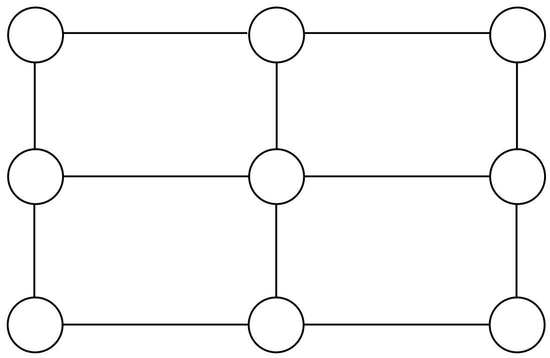
- Imagine that a time series is created from a set of M loosely-coupled underlying mechanisms
- Each of these mechanisms may have their own particular dynamic laws, and we want to avoid merging them into a single "meta-state," with a single transition matrix
  - avoid choosing a single time scale
  - avoid over-parameterization
- Here is the graphical model that we would like to use:



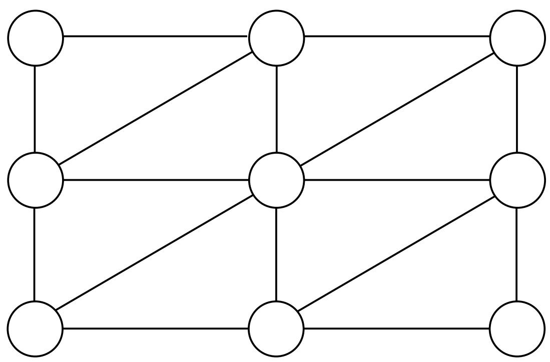
- When we triangulate do we get an efficient structure?

Triangulation?

- Here is the basic structure that we need to triangulate:



- Unfortunately, the following graph is not triangulated:



- To see this, "pick up" the central node. The "base" is a chordless cycle of length six.

- When we triangulate this kind of lattice structure we get an algorithm that scales as $O(N^6 T)$. The junction tree algorithm is not efficient for factorial HMMs.

Variational algorithms

- Later we will show how variational algorithms can be used to calculate *bounds* on the exact values of the probabilities
- The bounds can be calculated efficiently (cf. Ghahramani & Jordan, 1997):

Modeling J. S. Bach's chorales

Discrete event sequences:

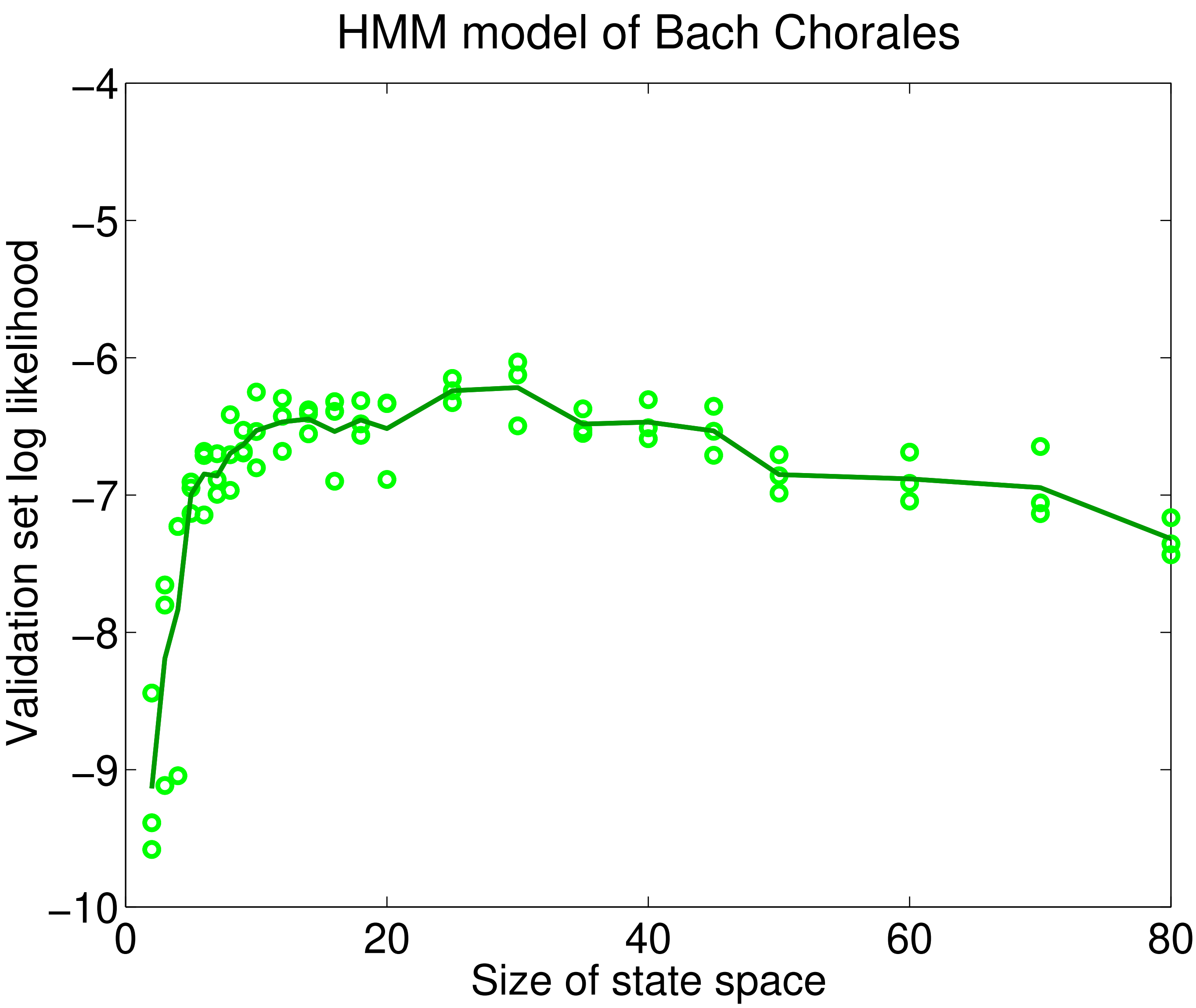
Attribute	Description	Representation
pitch	pitch of the event	int $[0, 127]$
keysig	key signature of the chorale (num of sharps and flats)	int $[7, 7]$
timesig	time signature of the chorale	int (1/16 notes)
fermata	event under fermata?	binary
st	start time of event	int (1/16 notes)
dur	duration of event	int (1/16 notes)

First 40 events of 66 chorale melodies:

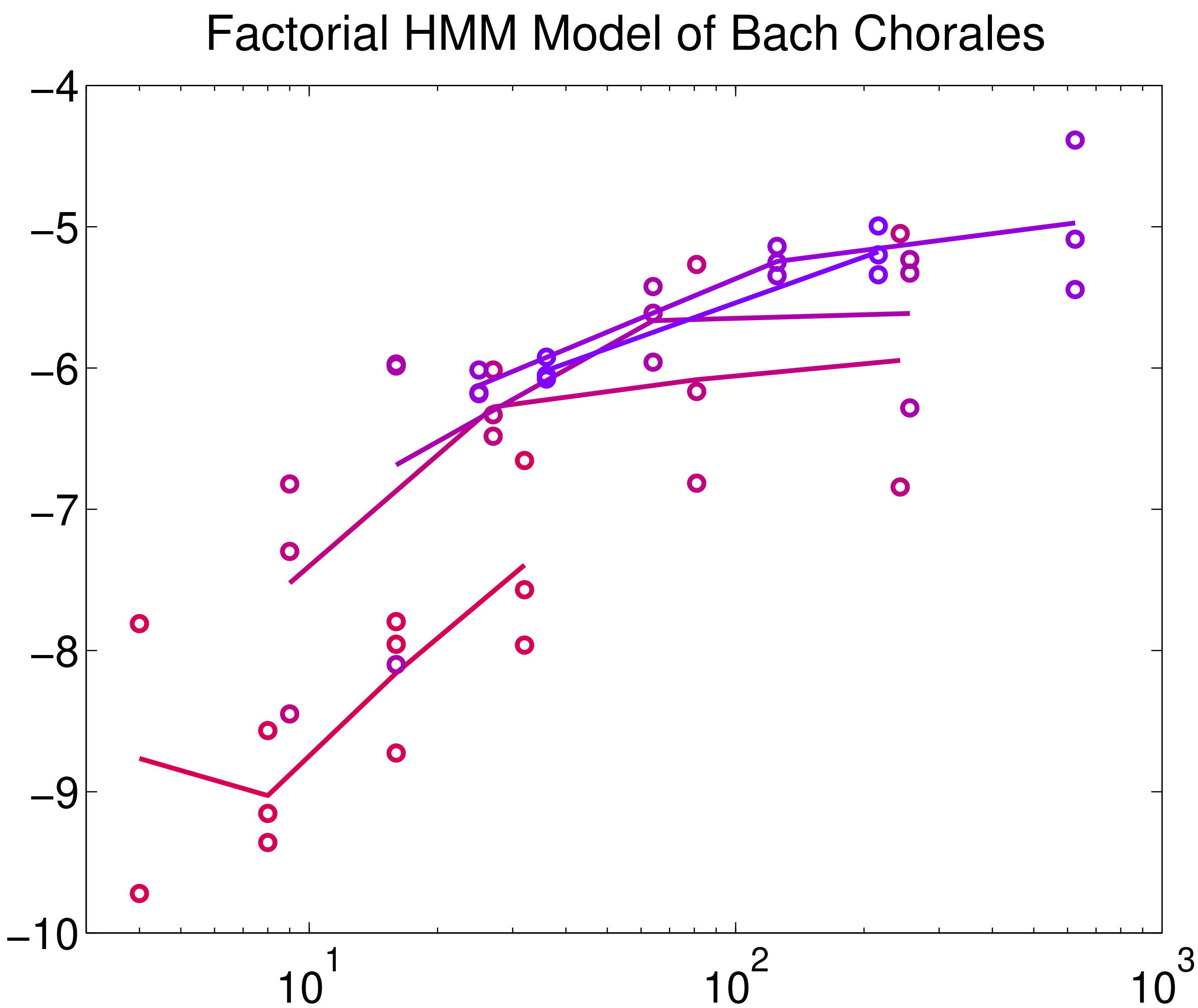
- training: 30 melodies
- test: 36 melodies

See Conklin and Witten (1995).

Fitting a single HMM to the Bach chorale data



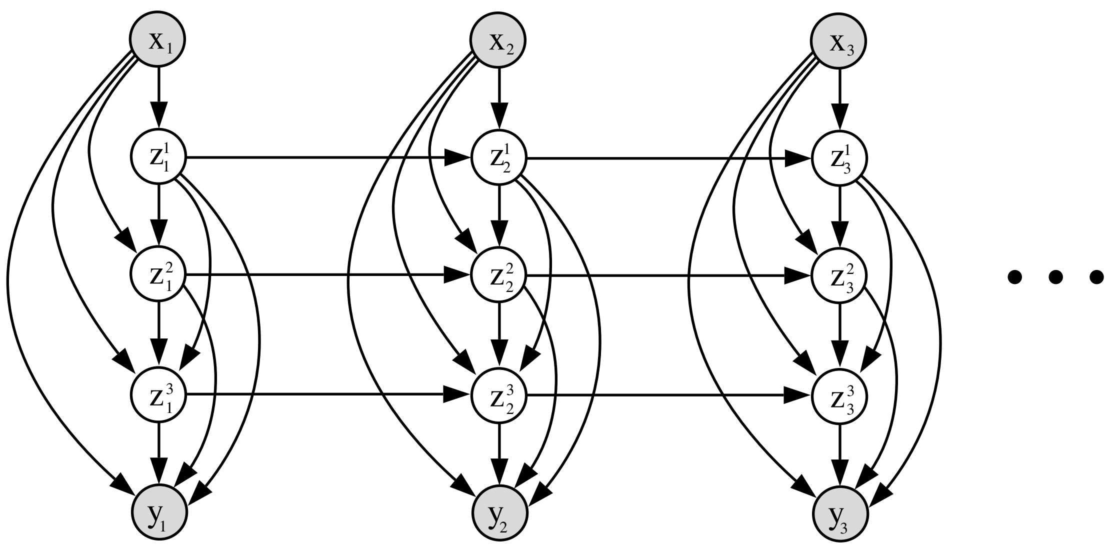
Fitting a factorial HMM to the Bach chorale data



Hidden Markov decision trees

(Jordan, Ghahramani, & Saul, 1997)

- We can combine decision trees with factorial HMMs
- This gives a "command structure" to the factorial representation

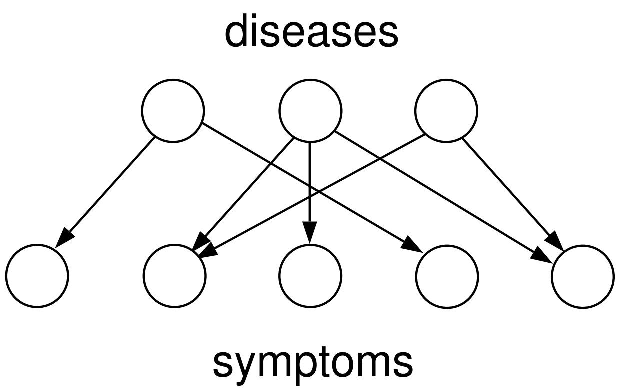


- Appropriate for multiresolution time series
- Again, the exact calculation is intractable and we must use variational methods

Quick Medical Reference (QMR)

(University of Pittsburgh)

- 600 diseases, 4000 symptoms
- arranged as a bipartite graph

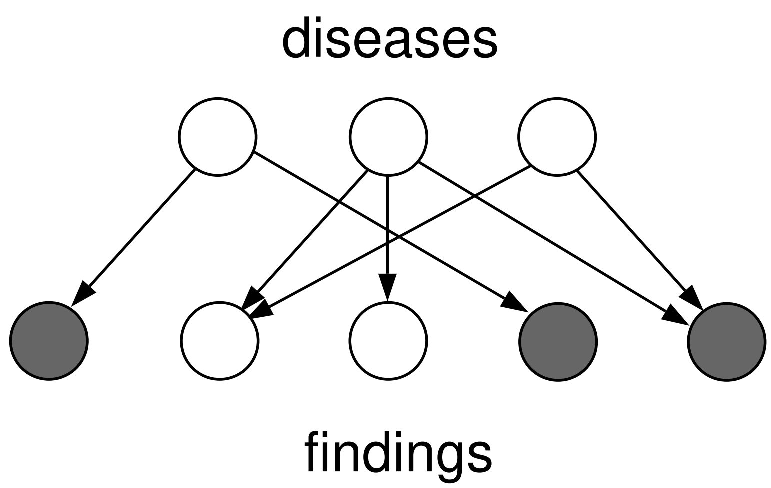


- Node probabilities $P(symptom_i|diseases)$ were obtained from an expert, under a noisy-OR model
- Want to do diagnostic calculations:

$$P(diseases|findings)$$

- Current methods (exact and Monte Carlo) are infeasible

QMR (cont.)



- "noisy-OR" parameterization:

$$P(f_i = 0|d) = (1 - q_{i0}) \prod_{j \in pa_i} (1 - q_{ij})^{d_j}$$

- rewrite in an exponential form:

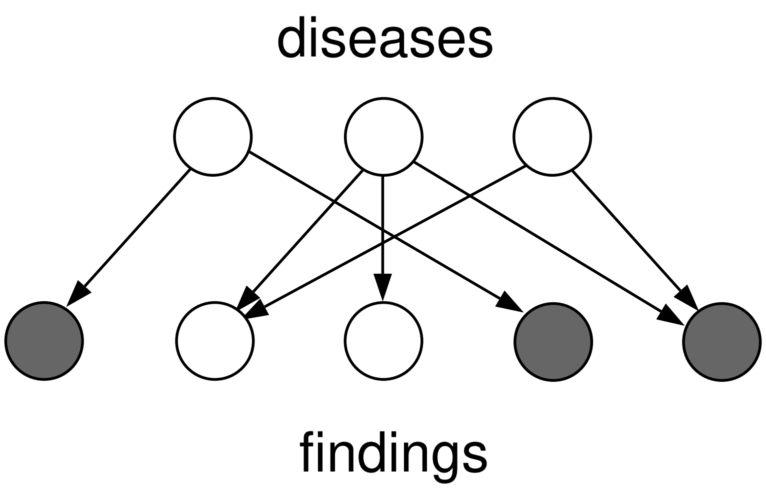
$$P(f_i = 0|d) = e^{-\theta_{i0} - \sum_{j \in pa_i} \theta_{ij} d_j}$$

where $\theta_{ij} \equiv -\log(1 - q_{ij})$

- probability of positive finding:

$$P(f_i = 1|d) = 1 - e^{-\theta_{i0} - \sum_{j \in pa_i} \theta_{ij} d_j}$$

QMR (cont.)



- Joint probability:

$$
\begin{aligned}
P(f,d) &= P(f|d)P(d) \\
&= \left[\prod_i P(f_i|d)\right]\left[\prod_j P(d_j)\right] \\
&= \left[\left(1-e^{-\theta_{10}-\sum_{j\in pa_1}\theta_{1j}d_j}\right)\left(1-e^{-\theta_{20}-\sum_{j\in pa_2}\theta_{2j}d_j}\right)\right. \\
&\qquad \left.\cdots\left(1-e^{-\theta_{k0}-\sum_{j\in pa_k}\theta_{kj}d_j}\right)\right]\left[\prod_j P(d_j)\right]
\end{aligned}
$$

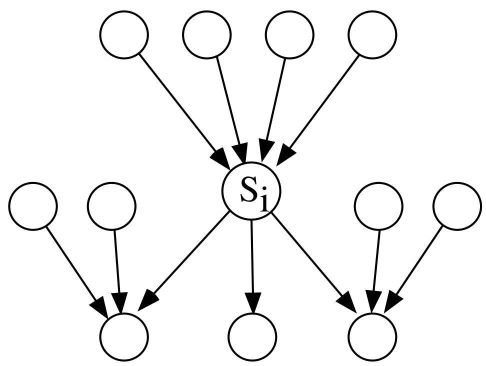
- Positive findings couple the disease nodes

Variational methods

- Variational methods are deterministic approximation methods
  - perhaps unique among deterministic methods in that they tend to work best for *dense* graphs
- They have some advantages compared to MCMC methods
  - they can be much faster
  - they yield upper and lower bounds on probabilities
- And they have several disadvantages
  - they are not as simple and widely applicable as MCMC methods
  - they require more art (thought on the part of the user) than MCMC methods
- But both variational approaches and MCMC approaches are evolving rapidly
- Note also that they can be combined (and can be combined with exact methods)

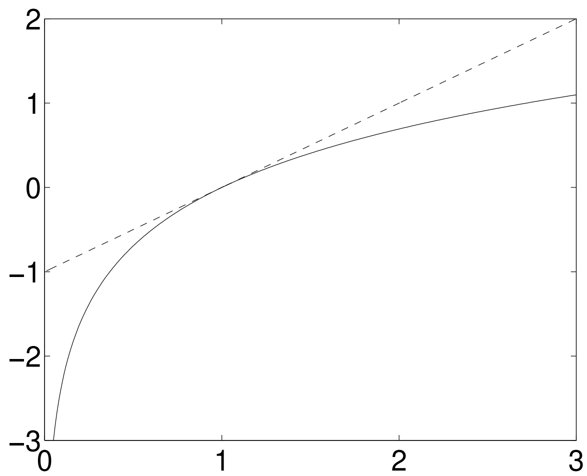
Introduction to variational methods

- Intuition—in a dense graph, each node is subject to many stochastic influences from nodes in its Markov blanket
  - laws of large numbers
  - coupled, nonlinear interactions between averages



- We want to exploit such averaging where applicable and where needed, while using exact inference algorithms on tractable "backbones"

Example of a variational transformation

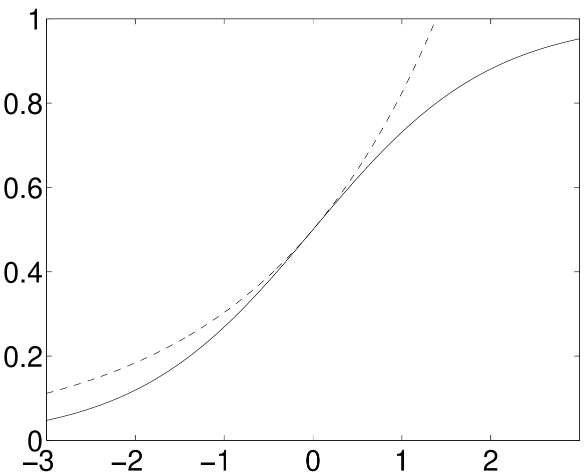


- $$\begin{aligned}\log(x) &= \min_{\lambda}\{\lambda x - \log\lambda - 1\} \\ &\leq \lambda x - \log\lambda - 1\end{aligned}$$

  – λ is a *variational parameter*

  – transforms a nonlinearity into a linearity

Example of a variational transformation



•

$$g(x) = \frac{1}{1+e^{-x}} \;=\; \min_{\lambda}\{e^{\lambda x - H(\lambda)}\}$$
$$\leq\; e^{\lambda x - H(\lambda)}$$

– where $H(\cdot)$ is the binary entropy

– a nonlinear function now a simple exponential

– cf. "tilted distributions"

Convex duality approach

(Jaakkola & Jordan, NIPS '97)

- for concave $f(x)$:

$$f(x) = \min_{\lambda} \left\{ \lambda^T x - f^*(\lambda) \right\}$$
$$f^*(\lambda) = \min_{x} \left\{ \lambda^T x - f(x) \right\}$$

- yields bounds:

$$f(x) \leq \lambda^T x - f^*(\lambda)$$
$$f^*(\lambda) \leq \lambda^T x - f(x)$$

- (lower bounds obtained from convex $f(x)$)
- cf. Legendre transformations

Variational transformations and inference

- Two basic approaches—*sequential* and *block*
  - we discuss the sequential approach first and return to the block approach later
- In the *sequential* approach, we introduce variational transformations sequentially, node by node
  - this yields a sequence of increasingly simple graphical models
  - every transformation introduces a new variational parameter, which we (lazily) put off evaluating until the end
  - eventually we obtain a model that can be handled by exact techniques, at which point we stop introducing transformations

Variational QMR

(Jaakkola & Jordan, 1997)

- Recall the noisy-OR probability of a positive finding:

$$P(f_i = 1|d) = 1 - e^{-\theta_{i0} - \sum_{j \in pa_i} \theta_{ij} d_j}$$

- The logarithm of this function is concave, thus we can utilize convex duality
  - evaluating the conjugate function, we obtain:

$$f^*(\lambda) = -\lambda \ln \lambda + (\lambda + 1) \ln(\lambda + 1)$$

- Thus we upper bound the probability of a positive finding:

$$\begin{aligned} P(f_i = 1|d) &\leq e^{\lambda_i(\theta_{i0} + \sum_{j \in pa_i} \theta_{ij} d_j) - f^*(\lambda_i)} \\ &= e^{\lambda_i \theta_{i0} - f^*(\lambda_i)} \prod_{j \in pa_i} \left[e^{\lambda_i \theta_{ij}} \right]^{d_j} \end{aligned}$$

- This is a factorized form; it effectively changes the "priors" $P(d_j)$ by multiplying them by $e^{\lambda_i \theta_{ij}}$ and delinking the i^{th} node from the graph

Variational QMR (cont.)

- Recall the joint distribution:

$$P(f,d) \;=\; \left[\left(1-e^{-\theta_{10}-\sum_{j\in pa_1}\theta_{1j}d_j}\right)\left(1-e^{-\theta_{20}-\sum_{j\in pa_2}\theta_{2j}d_j}\right)\right.$$
$$\left.\cdots\left(1-e^{-\theta_{k0}-\sum_{j\in pa_k}\theta_{kj}d_j}\right)\right]\left[\prod_j P(d_j)\right]$$

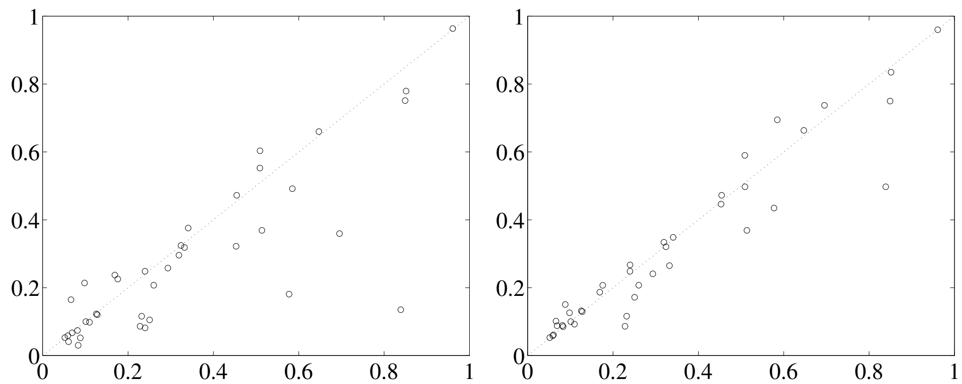
- After a variational transformation:

$$P(f,d) \;\leq\; \left[\left(1-e^{-\theta_{10}-\sum_{j\in pa_1}\theta_{1j}d_j}\right)\left(1-e^{-\theta_{20}-\sum_{j\in pa_2}\theta_{2j}d_j}\right)\right.$$
$$\left.\cdots\left(e^{\lambda_k\theta_{k0}-f^*(\lambda_k)}\prod_{j\in pa_k}\left[e^{\lambda_k\theta_{kj}}\right]^{d_j}\right)\right]\left[\prod_j P(d_j)\right]$$

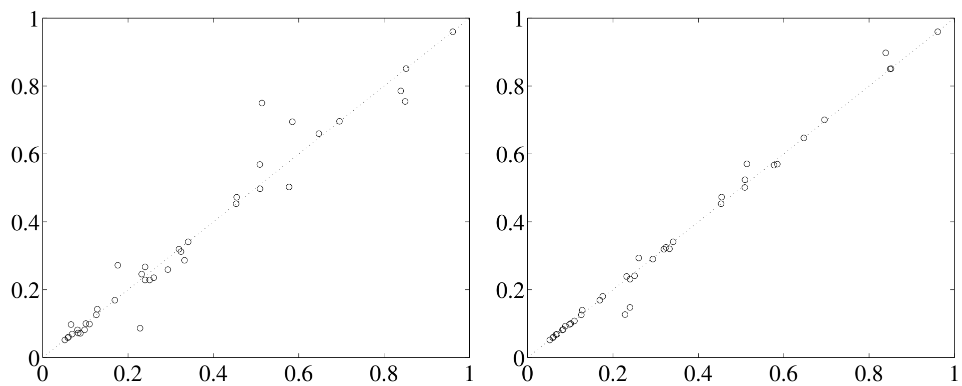
- Use a greedy method to introduce the variational transformations
  - choose the node that is estimated to yield the most accurate transformed posterior
- Optimize across the variational parameters λ_i
  - this turns out to be a convex optimization problem

QMR results

- We studied 48 clinicopathologic (CPC) cases
- 4 of these CPC cases had less than 20 positive findings; for these we calculated the disease posteriors exactly
- Scatterplots of exact vs. variational posteriors for (a) 4 positive findings treated exactly, (b) 8 positive findings treated exactly:

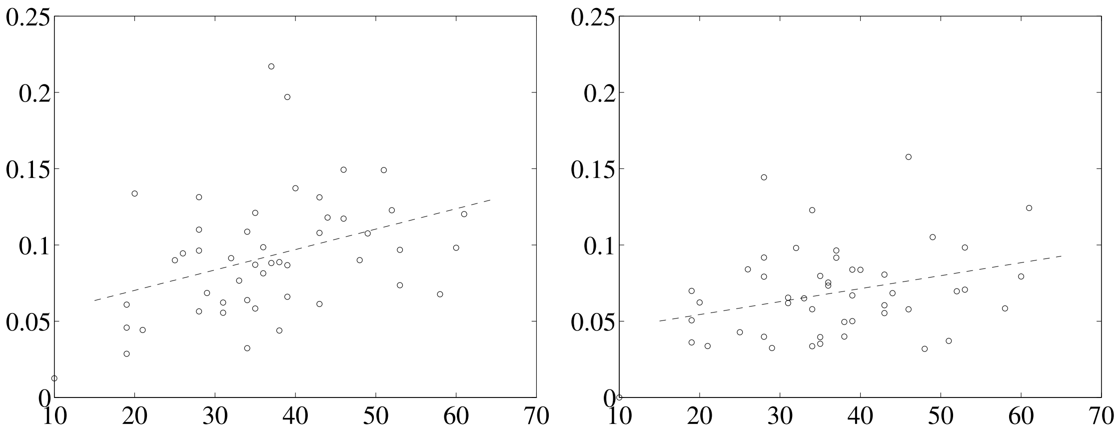


- Scatterplots of exact vs. variational posteriors for (a) 12 positive findings treated exactly, (b) 16 positive findings treated exactly:



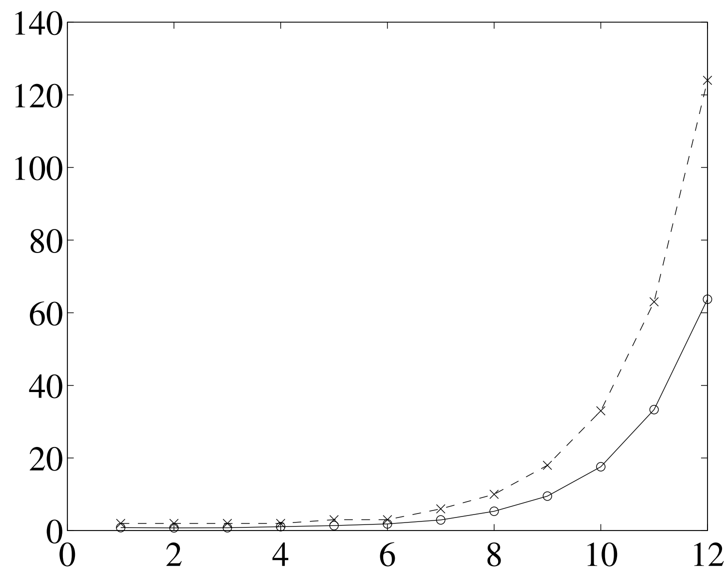
QMR results (cont.)

- For the remaining 44 CPC cases, we had no way to calculate the gold standard
- Thus we assessed the variational accuracy via a sensitivity estimate
  - we calculated the squared difference in posteriors when a particular node was treated exactly or variationally transformed, and averaged across nodes
  - a small value of this average suggests that we have the correct posteriors
  - (we validated this surrogate on the 4 cases for which we could do the exact calculation)
- Sensitivity estimates vs. number of positive findings when (a) 8 positive findings treated exactly, (b) 12 positive findings treated exactly:



QMR results (cont.)

- Timing results in seconds (Sparc 10) as a function of the number of positive findings treated exactly (solid line—average across CPC cases; dashed line—maximum across CPC cases):

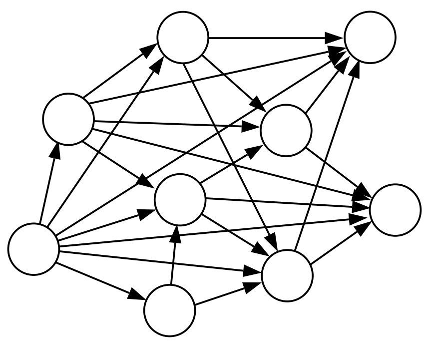


A cautionary note

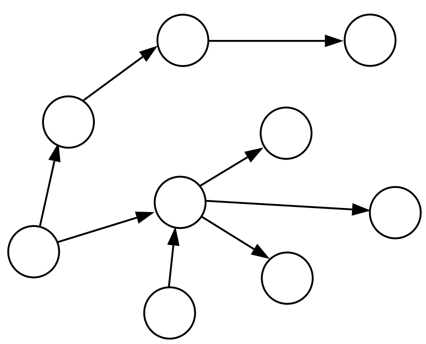
- these disease marginals are based on the upper variational distribution, which appears tight
- this distribution is guaranteed to upper bound the *likelihood*
- to obtain direct upper and lower bounds on the disease *marginals*, which are conditional probabilities, we need upper *and* lower bounds on the likelihood
- the lower bounds we obtain in our current implementation, however, are not tight enough
- thus, although the marginals we report appear to yield good approximations empirically, we cannot guarantee that they bound the true marginals

Variational transformations and inference

- Two basic approaches—*sequential* and *block*
- The block approach treats the approximation problem as a global optimization problem
- Consider a dense graph characterized by a joint distribution $P(H, V|\theta)$



- We remove links to obtain a simpler graph characterized by a conditional distribution $Q(H|V, \theta, \mu)$



- (Examples will be provided below...)

Variational inference (cont.)

- $Q(H|V, \theta, \mu)$ has extra degrees of freedom, given by *variational parameters* μ_i
  - we can think of these as being obtained by a sequence of variational transformations applied to the nodes
  - but we now want to take a more global view
- Choose μ_i so as to minimize

$$KL(Q\|P) = \sum_H Q(H|V, \theta, \mu) \log \frac{Q(H|V, \theta, \mu)}{P(H|V, \theta)}$$

- We will show that this yields a lower bound on the probability of the evidence (the likelihood)
- Minimizing the KL divergence requires us to compute averages under the Q distribution; we must choose Q so that this is possible
  - i.e., we choose our simplified graph so that it is amenable to exact methods

Variational inference (cont.)

- The fact that this is a lower bound follows from Jensen's inequality (cf. Hinton, et al., 1995):

$$\begin{aligned}\log P(V) &= \log \Sigma_H P(H,V) \\ &= \log \Sigma_H Q(H|V) \cdot \frac{P(H,V)}{Q(H|V)} \\ &\geq \Sigma_H Q(H|V) \log \left[\frac{P(H,V)}{Q(H|V)}\right]\end{aligned}$$

- The difference between the left and right hand side is the KL divergence:

$$KL(Q||P) = \Sigma_H Q(H|V) \log \left[\frac{Q(H|V)}{P(H|V)}\right]$$

which is positive; thus we have a lower bound

Linking the two approaches

(Jaakkola, 1997)

- The block approach can be derived within the convex duality framework

$$f(x) \geq \lambda^T x - f^*(\lambda)$$

  - treat the distribution $Q(H|V, \theta, \mu)$ as λ; a vector-valued variational parameter (one value for each configuration H)
  - the argument x becomes $\log P(H, V|\theta)$; also a vector-valued variable (one value for each configuration H)
  - the function $f(x)$ becomes $\log P(V|\theta)$
  - it turns out that the conjugate function $f^*(x)$ is the negative entropy function

- Thus convex duality yields:

$$\log P(V) \geq \sum_H Q(H|V) \log P(H, V) - \sum_H Q(H|V) \log Q(H|V)$$

  which is the bound derived earlier from Jensen's inequality

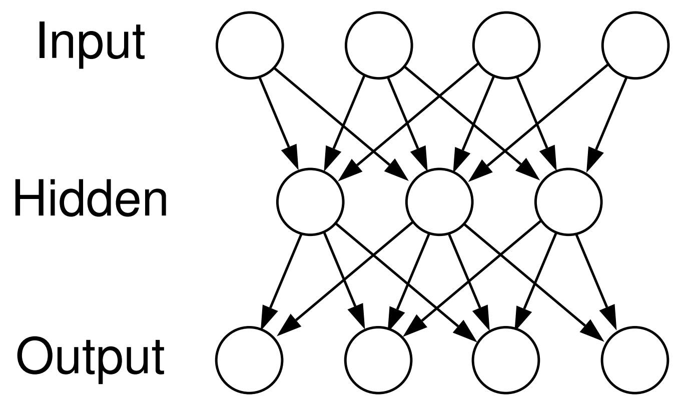
Learning via variational methods

- MAP parameter estimation for graphical models:
  - the EM algorithm is a general method for MAP estimation
  - "inference" is the E step of EM for graphical models (calculate $P(H|V)$ to "fill in" the hidden values)
  - variational methods provide an approximate E step
  - more specifically we increase the lower bound on the posterior at each iteration
- Bayesian methods for graphical models:
  - the variational approximation to the posterior may be easier to integrate than the true posterior
  - the variational approximation to the posterior may be easier to sample from than the true posterior

Neural networks and variational approximations

(Saul, Jaakkola, & Jordan, 1995)

- A multilayer neural network with logistic hidden units:



has a joint distribution that is a product of logistic functions:

$$P(H, V|\theta) = \Pi_{i \in (H,V)} \left[\frac{1}{1 + e^{-\sum_j \theta_{ij} S_j}} \right]$$

- The simplest variational approximation, which we will refer to as a *mean field* approximation considers the factorized approximation:

$$Q(H|V, \mu) = \Pi_{i \in H} \ \mu_i^{S_i} (1 - \mu_i)^{1-S_i}.$$

Division of labor

- The KL bound has two basic components:

$$\begin{pmatrix} \textit{variational} \\ \textit{entropy} \end{pmatrix} = -\Sigma_H \, Q(H|V) \log Q(H|V)$$

$$\begin{pmatrix} \textit{variational} \\ \textit{energy} \end{pmatrix} = -\Sigma_H \, Q(H|V) \log P(H,V)$$

- What we need is the difference:

$$\log P(V|\theta) \geq \begin{pmatrix} \textit{variational} \\ \textit{entropy} \end{pmatrix} - \begin{pmatrix} \textit{variational} \\ \textit{energy} \end{pmatrix}$$

Maximizing the lower bound

- We find the best approximation by varying $\{\mu_i\}$ to minimize $KL(Q||P)$.

- This amounts to maximizing the lower bound:

$$\log P(V|\theta) \geq \begin{pmatrix} \textit{variational} \\ \textit{entropy} \end{pmatrix} - \begin{pmatrix} \textit{variational} \\ \textit{energy} \end{pmatrix}.$$

- Abuse of notation: we define *clamped* values (0 or 1) for the instantiated nodes,

$$\mu_i = S_i \text{ for } i \in V.$$

Variational entropy

- The variational entropy is:

$$-\Sigma_H \; Q(H|V) \log Q(H|V)$$

- Our *factorized approximation* is:

$$Q(H|V) = \Pi_i \; \mu_i^{S_i} (1-\mu_i)^{1-S_i},$$

- The *joint entropy* is the sum of the individual unit entropies:

$$-\Sigma_i \; \left[\mu_i \log \mu_i + (1-\mu_i) \log(1-\mu_i) \right].$$

Variational energy

- The variational energy is:

$$-\Sigma_H \; Q(H|V) \log P(H,V)$$

- In sigmoid networks:

$$\begin{aligned} \log P(H,V) \;=\; & -\sum_{ij} \theta_{ij} S_i S_j \\ & + \sum_i \log\left[1 + e^{\Sigma_j \theta_{ij} S_j}\right]. \end{aligned}$$

- The first two terms are common to undirected networks. *The last term is not.*

- Averaging over $Q(H|V)$ gives:

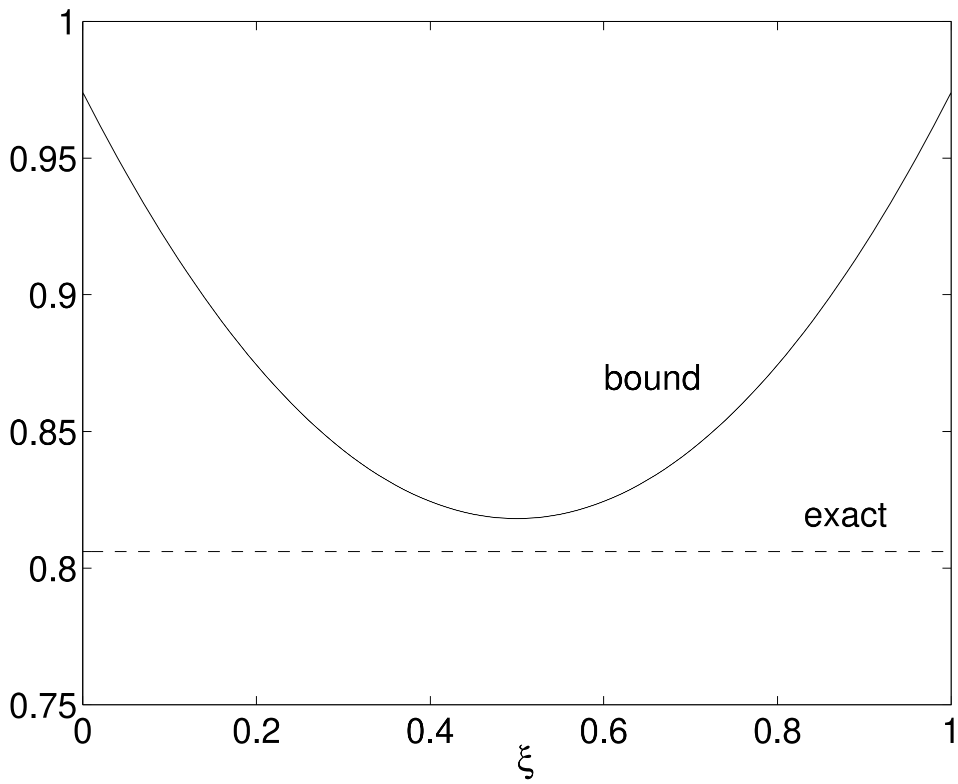
$$-\sum_{ij} \theta_{ij} \mu_i \mu_j \sum_i \left\langle \log\left[1 + e^{\Sigma_j \theta_{ij} S_j}\right]\right\rangle$$

- The last term is intractable, so (again) we use Jensen's inequality to obtain an upper bound…

- **Lemma** (Seung): for any random variable z, and any real number ξ:

$$\langle \log[1+e^z] \rangle \leq \xi \langle z \rangle + \log \left\langle e^{-\xi z} + e^{(1-\xi)z} \right\rangle.$$

- **Ex**: z is Gaussian with zero mean and unit variance.



- **Ex**: z is the sum of weighted inputs to S_i;

$$z = \sum_j \theta_{ij} S_j + h_i.$$

The lemma provides an upper bound on the intractable terms in the variational energy:

$$\sum_i \left\langle \log \left[1 + e^{\sum_j \theta_{ij} S_j} \right] \right\rangle$$

Variational (mean field) equations

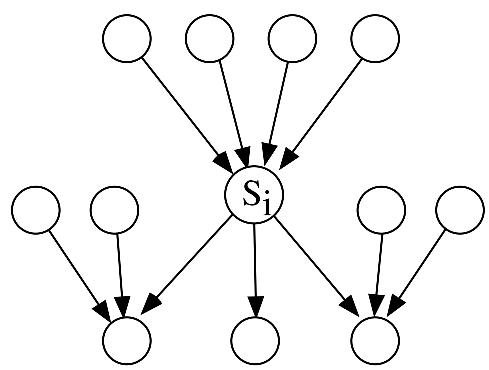
- The bound on the log-likelihood,

$$\log P(V|\theta) \geq \begin{pmatrix} \textit{variational} \\ \textit{entropy} \end{pmatrix} - \begin{pmatrix} \textit{variational} \\ \textit{energy} \end{pmatrix},$$

  is valid for any setting of the variational parameters, $\{\mu_i\}$

- The optimal $\{\mu_i\}$ are found by solving the variational equations:

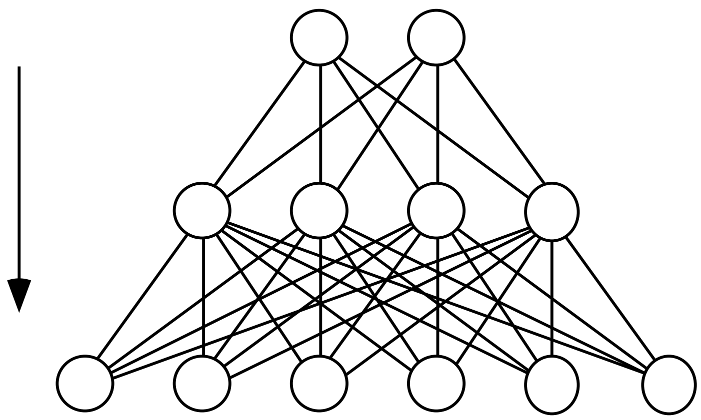
$$\mu_i = \sigma\left(\sum_j \left[\theta_{ij}\mu_j + \theta_{ji}(\mu_j - \xi_j) - K_{ji}\right]\right)$$



- The effective input to S_i is composed of terms from its *Markov blanket*

Numerical experiments

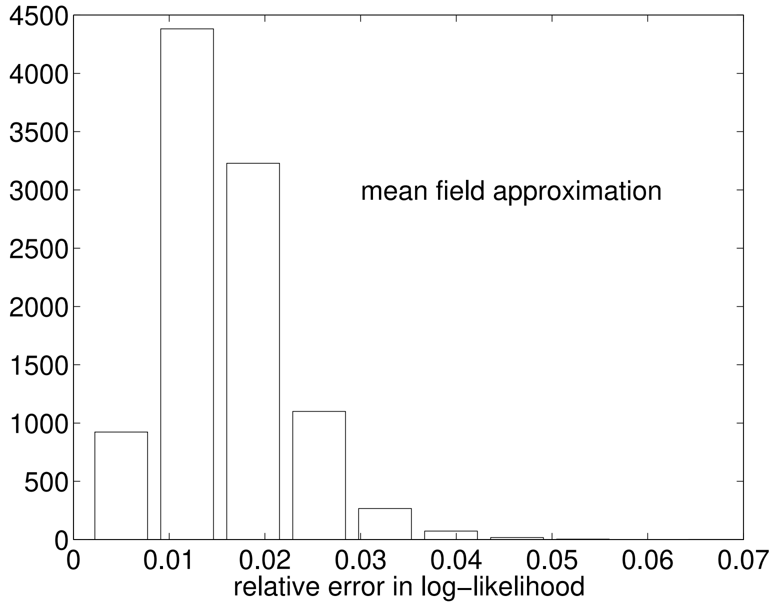
- For small networks, the variational bound can be compared to the true likelihood obtained by exact enumeration.



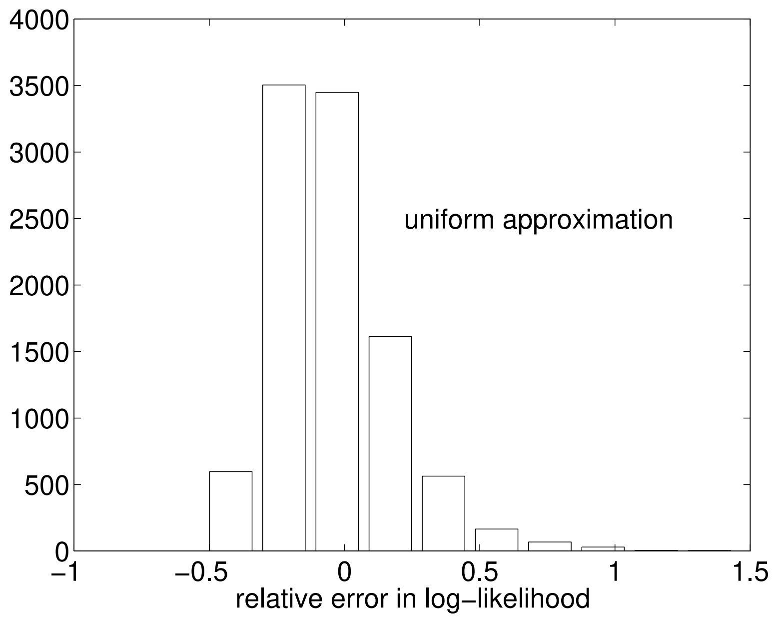
- We considered the event that all the units in the bottom layer were inactive.

- This was done for 10000 random networks whose weights and biases were uniformly distributed between -1 and 1.

- *Mean field approximation*:

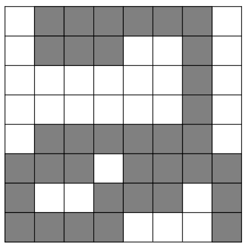
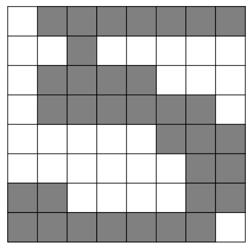


- *Uniform approximation*:



Digit recognition

- Images:

- Confusion matrix:

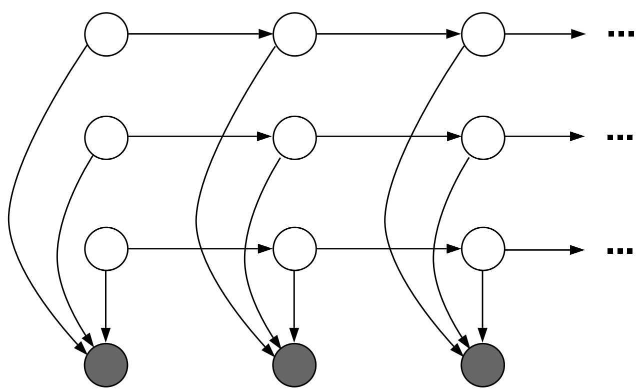
	0	1	2	3	4	5	6	7	8	9
0	388	2	2	0	1	3	0	0	4	0
1	0	393	0	0	0	1	0	0	6	0
2	1	2	376	1	3	0	4	0	13	0
3	0	2	4	373	0	12	0	0	6	3
4	0	0	2	0	383	0	1	2	2	10
5	0	2	1	13	0	377	2	0	4	1
6	1	4	2	0	1	6	386	0	0	0
7	0	1	0	0	0	0	0	388	3	8
8	1	9	1	7	0	7	1	1	369	4
9	0	4	0	0	0	0	0	8	5	383

- Comparative results:

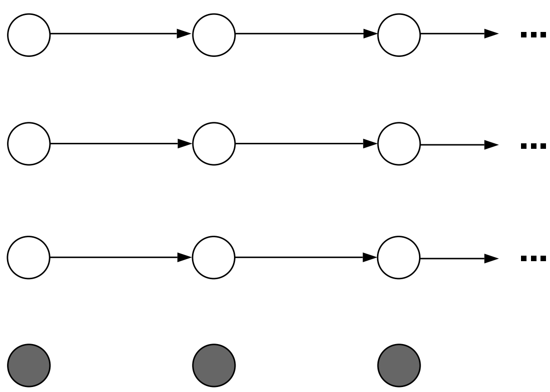
algorithm	test error
nearest neighbor	6.7
back propagation	5.6
Helmholtz machine	4.8
variational	4.6

Example—Factorial HMM

- Recall the factorial hidden Markov model, which yielded intractably large cliques when triangulated:

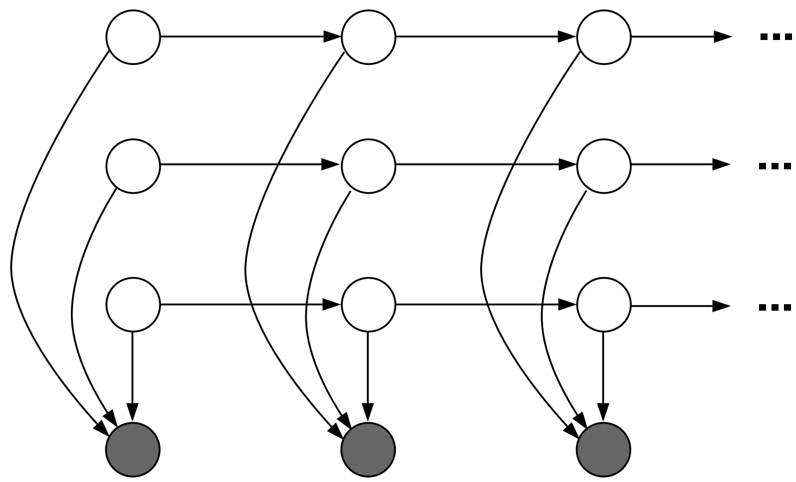


- We can variationally transform this model into:



where we see that we have to solve separate simple HMM problems on each iteration. The variational parameters couple the chains.

Example—Factorial HMM (cont.)



- M independent Markov chains given the observations

$$Q(\{S_t\}|\theta) = \prod_{m=1}^{M} Q(S_1^{(m)}|\theta) \prod_{t=2}^{T} Q(S_t^{(m)}|S_{t-1}^{(m)}, \theta),$$

  where

$$\begin{aligned} Q(S_1^{(m)}|\theta) &= \pi^{(m)} h_1^{(m)} \\ Q(S_t^{(m)}|S_{t-1}^{(m)}, \theta) &= P^{(m)} h_t^{(m)}. \end{aligned}$$

- The parameters of this approximation are the $h_t^{(m)}$, which play the role of observation log probabilities for the HMMs.

Example—Factorial HMM (cont.)

- Minimizing the KL divergence results in the following fixed point equation:

  $$h_t^{(m)} \propto \exp\{W^{(m)'}C^{-1}\left(Y_t - \hat{Y}_t\right) \; + \; W^{(m)'}C^{-1}W^{(m)}\langle S_t^{(m)}\rangle \\ - \; \frac{1}{2}\Delta^{(m)}\},$$

  where

  $$\hat{Y}^t = \sum_{\ell=1}^{M} W^{(\ell)}\langle S_t^{(\ell)}\rangle,$$

  and $\Delta^{(m)}$ is the vector of diagonal elements of $W^{(m)'}C^{-1}W^{(m)}$.

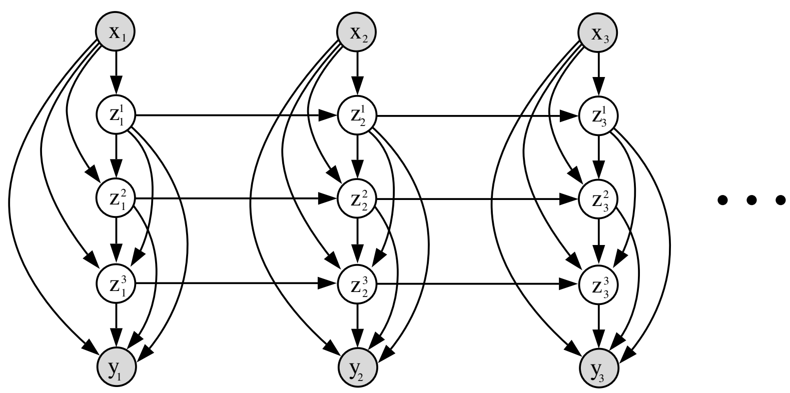
- Repeat until convergence of $KL(Q\|P)$:

  1. Compute $h_t^{(m)}$ using fixed-point equation, which depends on $\langle S_t^{(m)}\rangle$
  2. Compute $\langle S_t^{(m)}\rangle$ using forward-backward algorithm on HMMs with observation log probabilities given by $h_t^{(m)}$

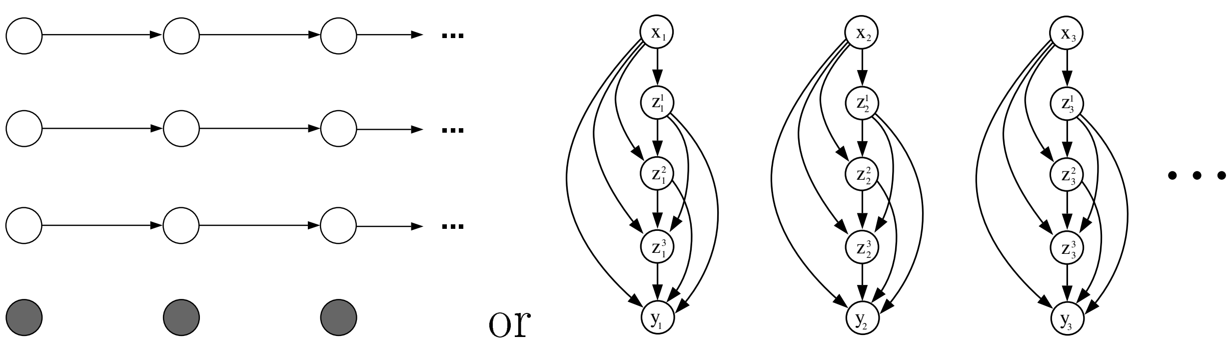
Example—hidden Markov decision tree

(Jordan, Ghahramani, & Saul, 1996)

- Recall the hidden Markov decision tree, which also yielded intractably large cliques when triangulated:

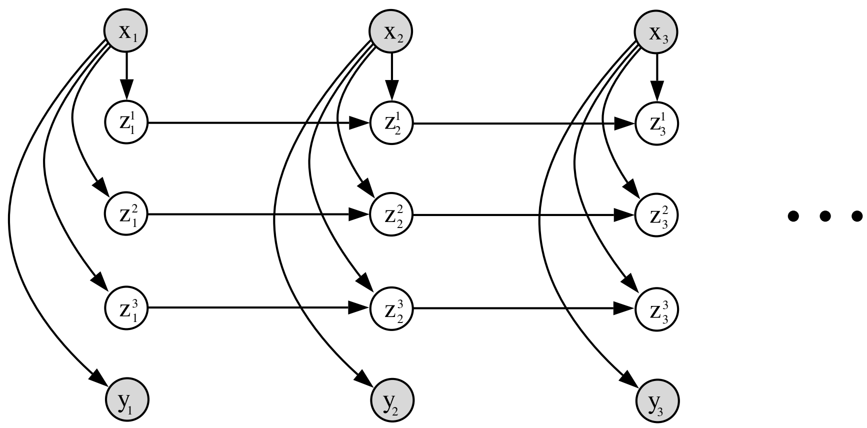


- We can variationally transform this model into one of two simplified models:



Forest of chains approximation

- Eliminating the vertical links that couple the states yields an approximating graph that is a forest of chains:



- The Q distribution is given by:

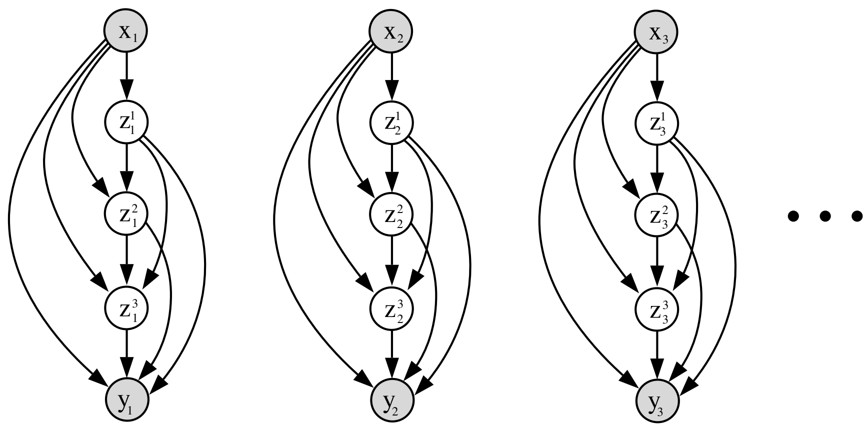
$$Q(\{\mathbf{z}_t^1, \mathbf{z}_t^2, \mathbf{z}_t^3\} \mid \{\mathbf{y}_t\}, \{\mathbf{x}_t\}) =$$
$$\frac{1}{Z_Q} \prod_{t=2}^{T} \tilde{a}_t^1(\mathbf{z}_t^1|\mathbf{z}_{t-1}^1)\tilde{a}_t^2(\mathbf{z}_t^2|\mathbf{z}_{t-1}^2)\tilde{a}_t^3(\mathbf{z}_t^3|\mathbf{z}_{t-1}^3)$$
$$\prod_{t=1}^{T} \tilde{q}_t^1(\mathbf{z}_t^1)\tilde{q}_t^2(\mathbf{z}_t^2)\tilde{q}_t^3(\mathbf{z}_t^3)$$

where $\tilde{a}_t^i(\mathbf{z}_t^i|\mathbf{z}_{t-1}^i)$ and $\tilde{q}_t^i(\mathbf{z}_t^i)$ are potentials that provide the variational parameterization

- The resulting algorithm is efficient because we know an efficient subroutine for single chains (the forward-backward algorithm)

Forest of trees approximation

- Eliminating the horizontal links that couple the states yields an approximating graph that is a forest of trees:



- The Q distribution is given by:

$$Q(\{\mathbf{z}_t^1, \mathbf{z}_t^2, \mathbf{z}_t^3\} \mid \{\mathbf{y}_t\}, \{\mathbf{x}_t\}) = \prod_{t=1}^{T} \tilde{r}_t^1(\mathbf{z}_1^1)\tilde{r}_t^2(\mathbf{z}_1^2|\mathbf{z}_1^1)\tilde{r}_t^3(\mathbf{z}_1^3|\mathbf{z}_1^1, \mathbf{z}_1^2)$$

- The resulting algorithm is efficient because we know an efficient subroutine for decision trees (the upward recursion from Jordan and Jacobs)

A Viterbi-like approximation

- We can develop a Viterbi-like algorithm by utilizing an approximation Q that assigns probability one to a single path $\{\bar{\mathbf{z}}_t^1, \bar{\mathbf{z}}_t^2, \bar{\mathbf{z}}_t^3\}$:

$$Q(\{\mathbf{z}_t^1, \mathbf{z}_t^2, \mathbf{z}_t^3\} \mid \{\mathbf{y}_t\}, \{\mathbf{x}_t\}) = \begin{cases} 1 & \text{if } \mathbf{z}_t^i = \bar{\mathbf{z}}_t^i, \quad \forall t, i \\ 0 & \text{otherwise} \end{cases}$$

- Note that the entropy $Q \ln Q$ is zero
- The evaluation of the energy $Q \ln P$ reduces to substituting $\bar{\mathbf{z}}_t^i$ for $\mathbf{z}_t^i$ in P
- The resulting algorithm involves a subroutine in which a standard Viterbi algorithm is run on a single chain, with the other (fixed) chains providing field terms

Mixture-based variational approximation

(Jaakkola & Jordan, 1997)

- Naive mean field approximations are unimodal
- Exact methods running as subroutines provide a way to capture multimodality
- But we would like a method that allows multimodality in the approximation itself

- This can be done by letting the Q distribution be a mixture:

$$Q_{mix}(H|V) = \sum_m \alpha_m Q_{mf}(H|V, m)$$

  where each component $Q_{mix}(H|V)$ is a factorized model

Mixture-based approximation (cont.)

- The bound on the likelihood takes the following form:

$$\mathcal{F}(Q_{mix}) = \sum_m \alpha_m \mathcal{F}(Q_{mf}|m) + I(m;H)$$

  where $I(m;H)$ is the mutual information between the mixture components

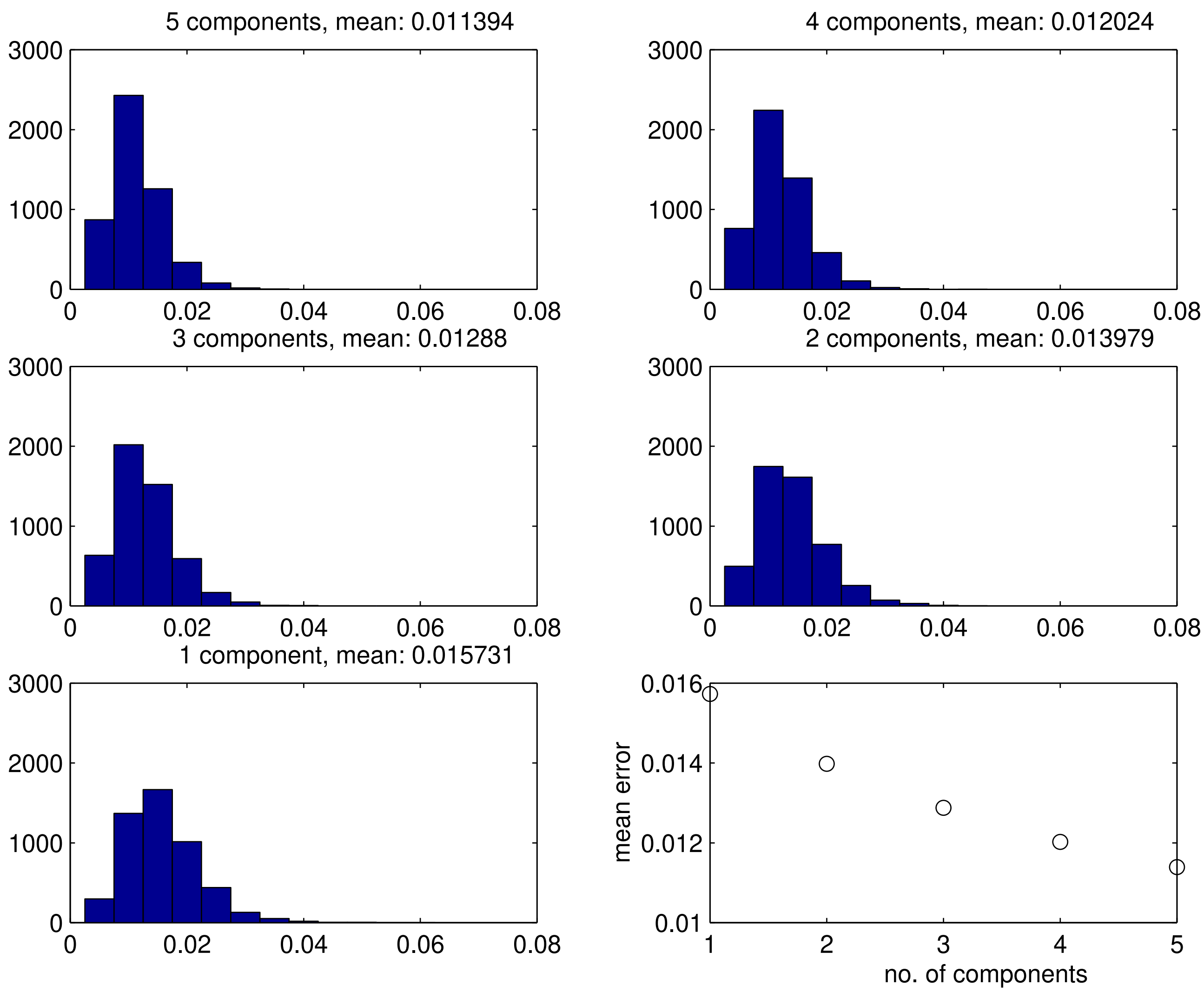
- (Derivation):

$$\begin{aligned}
\mathcal{F}(Q_{mix}) &= \sum_H Q_{mix}(H) \log \frac{P(H,V)}{Q_{mix}(H)} \\
&= \sum_{m,H} \alpha_m \left[Q_{mf}(H|m) \log \frac{P(H,V)}{Q_{mix}(H)} \right] \\
&= \sum_{m,H} \alpha_m \left[Q_{mf}(H|m) \log \frac{P(H,V)}{Q_{mf}(H|m)} + Q_{mf}(H|m) \log \frac{Q_{mf}(H|m)}{Q_{mix}(H)} \right] \\
&= \sum_m \alpha_m \mathcal{F}(Q_{mf}|m) + \sum_{m,H} \alpha_m Q_{mf}(H|m) \log \frac{Q_{mf}(H|m)}{Q_{mix}(H)} \\
&= \sum_m \alpha_m \mathcal{F}(Q_{mf}|m) + I(m;H)
\end{aligned}$$

- We see that the bound can be improved vis-a-vis naive mean field via the mutual information term

Mixture-based variational approximation

(Bishop, et al., 1997)



Conclusions

- General frameworks for probabilistic computation in graphical models:
  - exact computation
  - deterministic approximation
  - stochastic approximation
- Variational methods are deterministic approximation methods
  - they aim to take advantage of law-of-large-numbers phenomena in dense networks
  - they yield upper and lower bounds on desired probabilities
- The techniques can be combined to yield hybrid techniques which may well turn out to be better than any single technique

REFERENCES

Little, R. J. A., & Rubin, D. B. (1987). *Statistical Analysis with Missing Data.* New York: John Wiley.

Rabiner, L., (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*, 257-285.

Smyth, P., Heckerman, D., and Jordan, M. I. (1997). Probabilistic independence networks for hidden Markov probability models. *Neural Computation*, *9*, 227–270.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees.* Wadsworth International Group, Belmont, CA.

Jordan, M. and Jacobs, R. (1994). Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6:181–214.

Saul, L. K., and Jordan, M. I. (1995). Boltzmann chains and hidden Markov models. In G. Tesauro, D. S. Touretzky & T. K. Leen, (Eds.), *Advances in Neural Information Processing Systems 7*, MIT Press, Cambridge MA.

Jordan, M. I., & Bishop, C. (1997). Neural networks. In Tucker, A. B. (Ed.), *CRC Handbook of Computer Science*, Boca Raton, FL: CRC Press.

Neal, R. (1992). Connectionist learning of belief networks. *Artificial Intelligence*, *56*, pp 71-113.

Hinton, G. E., Dayan, P., Frey, B., and Neal, R. (1995). The wake-sleep algorithm for unsupervised neural networks. *Science*, 268:1158–1161.

Dayan, P., Hinton, G. E., Neal, R., and Zemel, R. S. (1995). Helmholtz Machines. *Neural Computation*, em 7, 1022-1037.

Saul, L., Jaakkola, T. and Jordan, M. (1996). Mean field theory for sigmoid belief networks. *Journal of Artificial Intelligence Research*, *4*, 61–76.

Ghahramani, Z., and Jordan, M. I. (1996). Factorial Hidden Markov models. In D. S. Touretzky, M. C. Mozer, & M. E. Hasselmo (Eds.), *Advances in Neural Information Processing Systems 8*, MIT Press, Cambridge MA.

Jaakkola, T., & Jordan, M. I. (1996). Computing upper and lower bounds on likelihoods in intractable networks. In E. Horvitz (Ed.), *Workshop on Uncertainty in Artificial Intelligence*, Portland, Oregon.

Frey, B., Hinton, G. E., & Dayan, P. (1996). Does the wake-sleep algorithm produce good density estimators? In D. S. Touretzky, M. C. Mozer, M. E. Hasselmo (Eds.), *Advances in Neural Information Processing Systems 8*, MIT Press, Cambridge MA.

Jaakkola, T., & Jordan, M. I. (1997). Bayesian logistic regression: a variational approach. In D. Madigan & P. Smyth (Eds.), Proceedings of the 1997 Conference on Artificial Intelligence and Statistics, Ft. Lauderdale, FL.

Saul, L. K., & Jordan, M. I. (1997). Mixed memory Markov models. In D. Madigan & P. Smyth (Eds.), Proceedings of the 1997 Conference on Artificial Intelligence and Statistics, Ft. Lauderdale, FL.

Jordan, M. I., Ghahramani, Z., & Saul, L. K. (1997). Hidden Markov decision trees. In M.

C. Mozer, M. I. Jordan, & T. Petsche (Eds.), *Advances in Neural Information Processing Systems 9*, MIT Press, Cambridge MA.

Jaakkola, T., & Jordan, M. I. (1997). Recursive algorithms for approximating probabilities in graphical models. In M. C. Mozer, M. I. Jordan, & T. Petsche (Eds.), *Advances in Neural Information Processing Systems 9*, MIT Press, Cambridge MA.

Jaakkola, T., & Jordan, M. I. (in press). Improving the mean field approximation via the use of mixture distributions. In M. I. Jordan (Ed.), *Learning and Inference in Graphical Models*, Kluwer Academic Press.

Bishop, C., Lawrence, N., Jaakkola, T. S., & Jordan, M. I. (in press). Approximating posterior distributions in belief networks using mixtures. *Advances in Neural Information Processing Systems 10.*