Probabilistic Graphical Models

Part Two: Inference and Learning

Christopher M. Bishop

Microsoft Research, Cambridge, U.K.

research.microsoft.com/~cmbishop

NATO ASI: Learning Theory and Practice, Leuven, July 2002

Overview of Part Two

- Exact inference and the junction tree
- MCMC
- Variational methods and EM
- Example
- General variational inference engine

Overview of Part Two

- Exact inference and the junction tree
- MCMC
- Variational methods and EM
- Example
- General variational inference engine

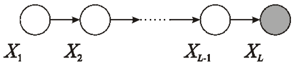
Exact Inference

- Group the hidden variables H into H_1 and H_2 in which we want to marginalize over H_2 to find the posterior over H_1
- Thus our most general inference problem involves evaluation of

$$P(H_1|V) = \sum_{H_2} P(H_1, H_2|V)$$

- For a M-state discrete units there are $M^{|H_2|}$ terms in the summation where $|H_2|$ is the number of hidden nodes
- Can easily become computationally intractable
- Can we exploit the conditional independence structure (missing links) to find more efficient algorithms?

Example



$$P(X_1, \ldots, X_L) = P(X_1)P(X_2|X_1)\ldots P(X_L|X_{L-1})$$

- Goal: find $P(X_1|X_L)$
- Direct evaluation gives

$$P(X_1|X_L) = \frac{\sum_{X_2} \cdots \sum_{X_{L-1}} P(X_1, \ldots, X_L)}{\sum_{X_1} \sum_{X_2} \cdots \sum_{X_{L-1}} P(X_1, \ldots, X_L)}$$

where, for variables having M states, the denominator involves summing over M^{L-1} terms (exponential in the length of the chain)

Example (cont'd)

- Using the conditional independence structure we can re-order the summations in the denominator to give

$$\sum_{X_{L-1}} P(X_L|X_{L-1}) \cdots \sum_{X_2} P(X_3|X_2) \sum_{X_1} P(X_2|X_1)P(X_1)$$

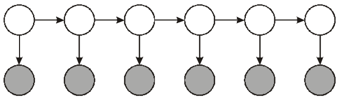
which involves $\sim LM^2$summations (linear in the length of the chain) – similarly for the numerator

- Can be viewed as a local message passing algorithm

Belief Propagation

- Extension to general tree structured graphs
- Involves passing one message in each direction across every link
- Exact solution in time linear in size of graph

Example: Hidden Markov Model



- Inference involves one forward and one backward pass
- Computational cost grows linearly with length of chain
- Similarly for the Kalman filter

Junction Tree Algorithm

- An efficient exact algorithm for a general graph
- Applies to both directed and undirected graphs
- Compiles the original graph into a tree structure and then performing message passing on this tree

Junction Tree Algorithm (cont'd)

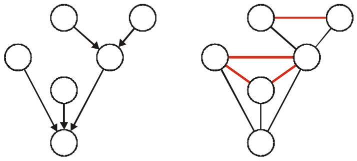
- Key steps:
  1. Moralize
  2. Absorb evidence
  3. Triangulate
  4. Construct junction tree of cliques
  5. Pass messages to achieve consistency

Moralization

- There are algorithms which work with the original directed graph, but these turn out to be special cases of the junction tree algorithm
- In the JT algorithm we first convert the directed graph into an undirected graph – directed and undirected graphs are then treated using the same approach
- Suppose we are given a directed graph with a conditionals $P(X_i|\mathrm{pa}_i)$ and we wish to find a representation as an undirected graph

Moralization (cont'd)

- The conditionals $P(X_i|\mathrm{pa}_i)$ are obvious candidates as clique potentials, but we need to ensure that each node belongs in the same clique as its parents
- This is achieved by adding, for each node, links connecting together all of the parents



Moralization (cont'd)

- Moralization therefore consists of the following steps:
  1. For each node in the graph, add an edge between all parents of the node and then convert directed edges to undirected edges
  2. Initialize the clique potentials of the moral graph to 1
  3. For each local conditional probability $P(X_i|\mathrm{pa_i})$ choose a clique C such that C contains both X_i and pa$_i$ and multiply ψ_C by $P(X_i|\mathrm{pa_i})$
- Note that this undirected graph automatically has a normalization factor $Z = 1$

Moralization (cont'd)

- By adding links we have discarded some conditional independencies
- However, any conditional independencies in the moral graph also hold for the original directed graph, so if we solve the inference problem for the moral graph we will solve it also for the directed graph

Absorbing Evidence

- The nodes can be grouped into visible V for, which we have particular observed values $V = v$, and hidden H
- We are interested in the conditional (posterior) probability

$$\begin{aligned} P(H|V=v) &= \frac{P(H,V=v)}{P(V=v)} \\ &= \frac{\prod_C \psi_C(H,V=v)}{\sum_H \prod_C \psi_C(H,V=v)} \end{aligned}$$

- Absorb evidence simply by altering the clique potentials to be zero for any configuration inconsistent with $V = v$

Absorbing Evidence (cont'd)

- We can view

$$\prod_C \psi_C(H,V=v)$$

as an un-normalized version of

$$P(H,V=v)$$

and hence an un-normalized version of

$$P(H|V=v)$$

Local Consistency

- As it stands, the graph correctly represents the (un-normalized) joint distribution $P(H,V=v)$ but the clique potentials do not have an interpretation as marginal probabilities
- Our goal is to update the clique potentials so that they acquire a local probabilistic interpretation while preserving the global distribution

Local Consistency (cont'd)

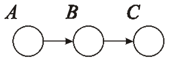
- Note that we cannot simply have

$$P(X) = \prod_C \psi_C(X_C)$$

with

$$\psi_C(X_C) = P(X_C)$$

as can be seen by considering the three node graph



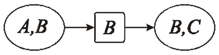
- Here

$$P(A,B,C) \neq P(A,B)P(B,C)$$

Local Consistency (cont’d)

- Instead we consider the more general representation for undirected graphs including separator sets, then we have

$$\begin{aligned} P(A,B,C) &= P(A)P(B|A)P(C|B) \\ &= \frac{P(A,B)P(B,C)}{P(B)} \end{aligned}$$



Local Consistency (cont’d)

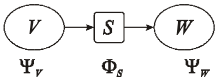
- Starting from our un-normalized representation of $P(H, V = v)$ in terms of products of clique potentials, we can introduce separator potentials initially set to unity

$$P(H, V = v) = \frac{1}{Z}\frac{\prod_C \psi_C(X_C)}{\prod_S \phi_S(X_S)}$$

  Note that nodes can appear in more than one clique, and we require that these be consistent for all marginals
- Achieving consistency is central to the junction tree algorithm

Local Consistency (cont’d)

- Consider the elemental problem of achieving consistency between a pair of cliques V and W, with separator set S



- Initially $\Phi_S = 1$

Local Consistency (cont’d)

- First construct a “message” at clique V and pass to W

$$\begin{aligned} \phi_S^\star &= \sum_{V\backslash S} \psi_V \\ \psi_W^\star &= \frac{\phi_S^\star}{\phi_S}\psi_W \end{aligned}$$

- Since ψ_V is unchanged $\psi_V^\star = \psi_V$, and so the joint distribution is invariant

$$\frac{\psi_V^\star \psi_W^\star}{\phi_S^\star} = \frac{\psi_V \psi_W}{\phi_S}$$

Local Consistency (cont’d)

- Next pass a message back from W to V using the same update rule

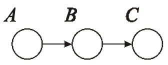
$$\begin{aligned} \phi_S^{\star\star} &= \sum_{W\backslash S} \psi_W^\star \\ \psi_V^{\star\star} &= \frac{\phi_S^{\star\star}}{\phi_S^\star}\psi_V^\star \end{aligned}$$

- Here $\psi_W^\star$ is unchanged and so $\psi_W^{\star\star} = \psi_W^\star$, and again the joint distribution is unchanged
- The marginals are now correct for both of the cliques and also for the separator

$$\sum_{V\backslash S} \psi_V^{\star\star} = \sum_{V\backslash S} \frac{\phi_S^{\star\star}}{\phi_S^\star}\psi_V^\star = \frac{\phi_S^{\star\star}}{\phi_S^\star}\sum_{V\backslash S} \psi_V^\star = \frac{\phi_S^{\star\star}}{\phi_S^\star}\phi_S^\star = \sum_{W\backslash S} \psi_W^{\star\star}$$

Local Consistency (cont’d)

- Example: return to the earlier three node graph



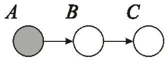
- Initially the clique potentials are $\psi_{AB} = P(A, B)$ and $\psi_{BC} = P(B|C)$, and the separator potential $\phi_B = 1$
- The first message pass gives the following update

$$\begin{aligned} \phi_B^\star &= \sum_A P(A,B) = P(B) \\ \psi_{BC}^\star &= \frac{P(B)}{1}P(C|B) = P(B,C) \end{aligned}$$

  which are the correct marginals
- In this case the second message is vacuous

Local Consistency (cont'd)

- Now suppose that node A is observed (for simplicity consider nodes to be binary and suppose A=1)



- Absorbing the evidence involves altering the potential ψ_{AB} by setting the A=0 row to zero
- Summing over A gives $\phi_B^\star = P(A = 1, B)$
- Updating the $\{B, C\}$ potential gives

$$\psi_{BC}^\star = \frac{P(A=1,B)}{1} P(C|B) = P(A=1,B,C)$$

Local Consistency (cont'd)

- Hence the potentials after the first message pass are

$$\begin{aligned} \psi_{AB}^\star &= P(A=1,B) \\ \phi_B^\star &= P(A=1,B) \\ \psi_{BC}^\star &= P(A=1,B,C) \end{aligned}$$

- Again the reverse message is vacuous
- Note that the resulting clique and separator marginals require normalization (a local operation)

Global Consistency

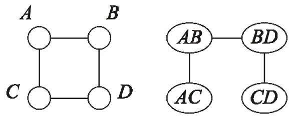
- How can we extend our two-clique procedure to ensure consistency across the whole graph?
- We construct a *clique tree* by considering a spanning tree linking all of the cliques which is maximal with respect to the cardinality of the intersection sets
- Next we construct and pass messages using the following protocol:
  - *a clique can send a message to a neighbouring clique only when it has received messages from all of its neighbours*

Global Consistency (cont'd)

- In practice this can be achieved by designating one clique as root and then
  - (i) collecting evidence by passing messages from the leaves to the root
  - (ii) distributing evidence by propagating outwards from the root to the leaves

One Last Issue

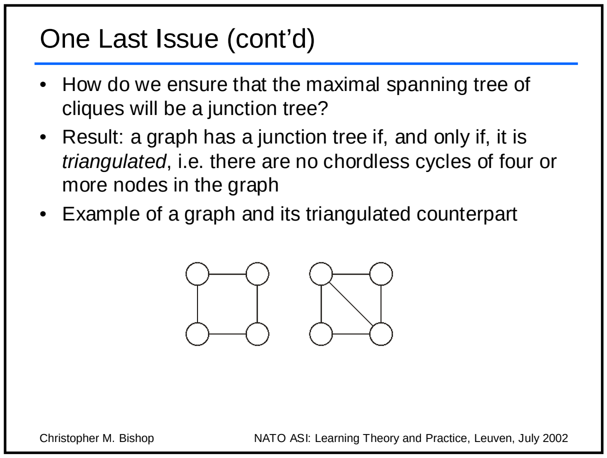
- The algorithm discussed so far is not quite sufficient to guarantee consistency for an arbitrary graph
- Consider the four node graph here, together with a maximal spanning clique tree



- Node C appears in two places and there is no guarantee that local consistency for $P(C)$ will result in global consistency

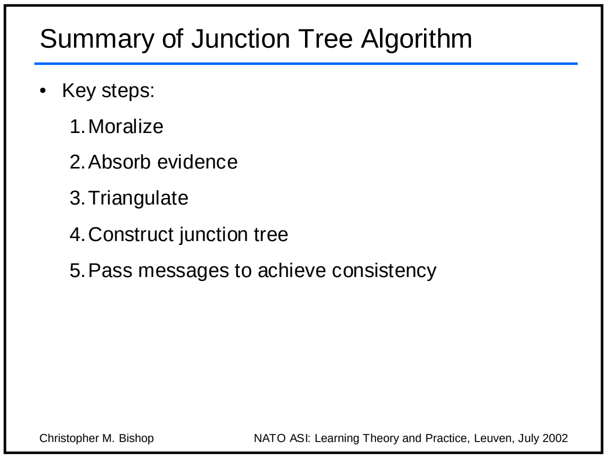
One Last Issue (cont'd)

- The problem is resolved if the tree of cliques is a *junction tree*, i.e. if for every pair of cliques V and W all cliques on the (unique) path from V to W contain $V \cap W$ (running intersection property)
- As a by-product we are also guaranteed that the (now consistent) clique potentials are indeed marginals



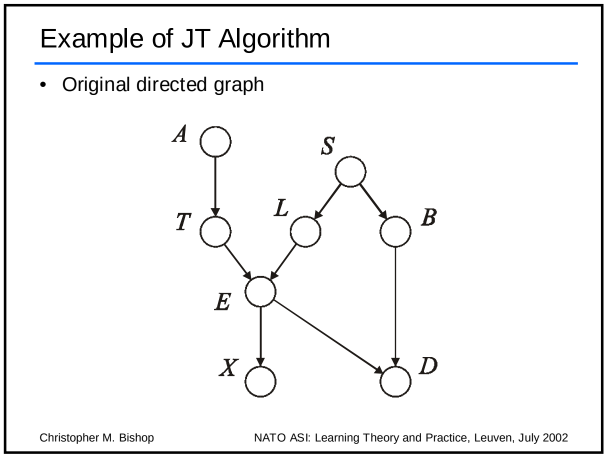
One Last Issue (cont'd)

- How do we ensure that the maximal spanning tree of cliques will be a junction tree?
- Result: a graph has a junction tree if, and only if, it is *triangulated*, i.e. there are no chordless cycles of four or more nodes in the graph
- Example of a graph and its triangulated counterpart



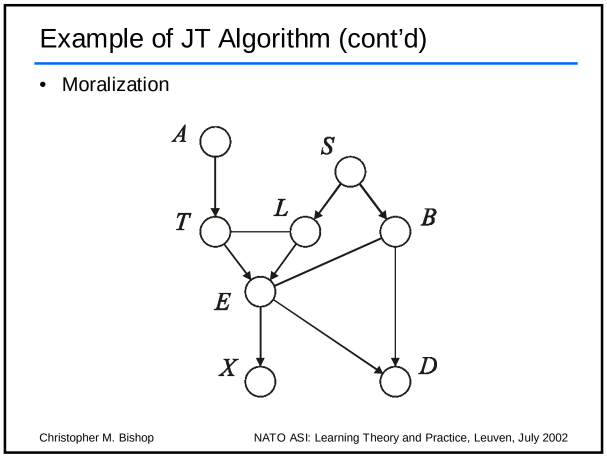
Summary of Junction Tree Algorithm

- Key steps:
  1. Moralize
  2. Absorb evidence
  3. Triangulate
  4. Construct junction tree
  5. Pass messages to achieve consistency



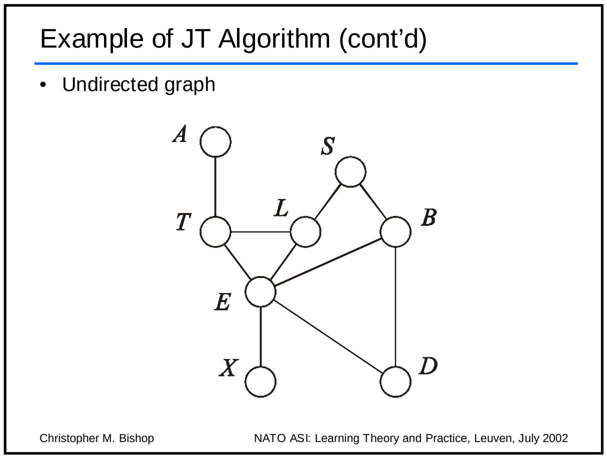
Example of JT Algorithm

- Original directed graph



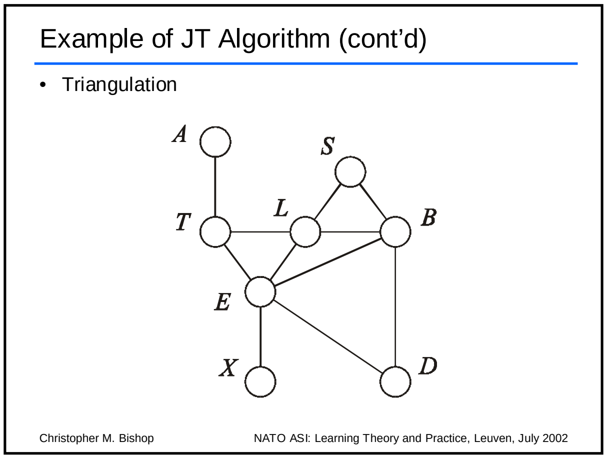
Example of JT Algorithm (cont'd)

- Moralization



Example of JT Algorithm (cont'd)

- Undirected graph

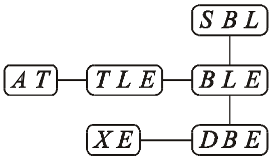


Example of JT Algorithm (cont'd)

- Triangulation

Example of JT Algorithm (cont'd)

- Junction tree



Limitations of Exact Inference

- The computational cost of the junction tree algorithm is determined by the size of the largest clique
- For densely connected graphs exact inference may be intractable
- There are 3 widely used approximation schemes
  - Pretend graph is a tree: "loopy belief propagation"
  - Markov chain Monte Carlo (e.g. Gibbs sampling)
  - Variational inference

Overview of Part Two

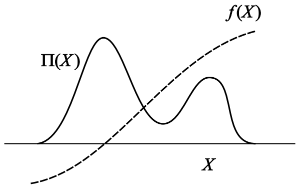
- Exact inference and the junction tree
- MCMC
- EM algorithm
- Variational methods
- Example
- General variational inference engine

MCMC

- Eventual goal is to evaluate averages of functions

$$\langle f \rangle = \int f(X)\Pi(X)\,dX$$

where $\Pi(X)$ is typically the posterior distribution of (some subset of) the hidden variables



MCMC (cont'd)

- Sampling methods aim to draw a sample of K points X_k from the distribution $\Pi(X)$ and then to approximate the expectation by a finite sum

$$\hat{f} = \frac{1}{K}\sum_{k=1}^{K} f(X_k)$$

- This has the correct mean $\langle \hat{f} \rangle = \langle f \rangle$ and variance

$$\frac{1}{K}\langle (f - \langle f \rangle)^2 \rangle$$

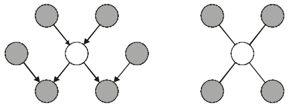
which is independent of dimensionality
- We can achieve excellent accuracy for small values of K
- However, this assumes sample points are independent!

MCMC (cont'd)

- *Gibbs sampling* samples each variable in turn using its conditional distribution conditioned on the other variables

$$P(X_i | X_{\{j \neq i\}})$$

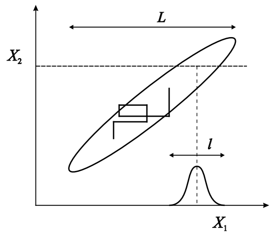
- Not limited to conjugate-exponential distributions
- For graphical models, these conditional distributions depend only on the variables in the *Markov blanket*



- Software implementation: BUGS (Spiegelhalter *et al.*)

MCMC (cont'd)

- The problem with Gibbs sampling is that successive points are highly correlated



- In this example it takes of order $(L/l)^2$ steps to generate independent samples (random walk)

Overview of Part Two

- Exact inference and the junction tree
- MCMC
- Variational methods and EM
- Example
- General variational inference engine

Learning in Graphical Models

- Introduce parameters $\theta = (\theta_1, \ldots, \theta_d)$ which govern the conditional distributions $P(X_i|\text{pa}_i, \theta_i)$ in a directed graph, or clique potentials $\psi_C(X_C; \theta_C)$ in an undirected graph
- Maximum likelihood: determine θ_{ML} by

$$\begin{aligned}\theta_{\text{ML}} &= \arg\max_{\theta} \{\ln P(V|\theta)\} \\ &= \arg\max_{\theta} \left\{\ln \sum_{H} P(V, H|\theta)\right\}\end{aligned}$$

- Problem: the summation over H inside the logarithm may be intractable

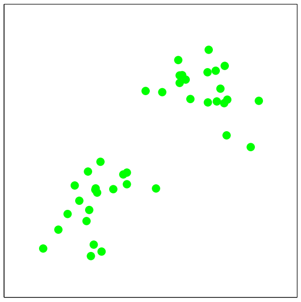
Expectation-maximization (EM) Algorithm

- E-step: evaluate the posterior distribution $P(H|V, \theta_{\text{old}})$ using current estimate θ_{old} for the parameters
- M-step: re-estimate θ by maximizing the expected *complete-data* log likelihood

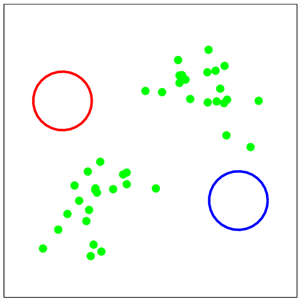
$$\theta_{\text{new}} = \arg\max_{\theta} \left\{\sum_{H} P(H|V, \theta_{\text{old}}) \ln P(V, H|\theta)\right\}$$

- Note that the log and the summation have been exchanged – this will often make the summation tractable
- Iterate E and M steps until convergence

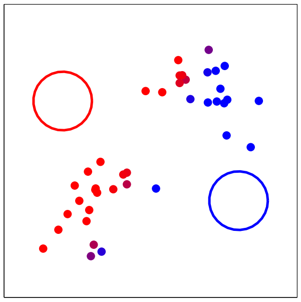
EM Example: Mixtures of Gaussians



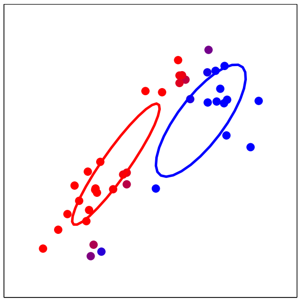
EM Example: Mixtures of Gaussians (cont'd)



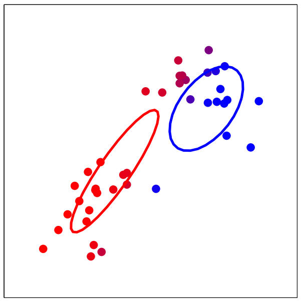
EM Example: Mixtures of Gaussians (cont'd)



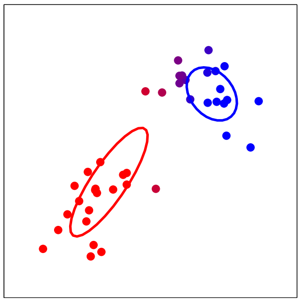
EM Example: Mixtures of Gaussians (cont'd)



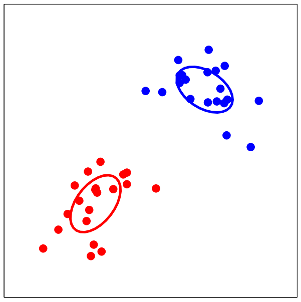
EM Example: Mixtures of Gaussians (cont'd)



EM Example: Mixtures of Gaussians (cont'd)



EM Example: Mixtures of Gaussians (cont'd)



EM: Variational Viewpoint

- For an arbitrary distribution $Q(H|V)$ we have

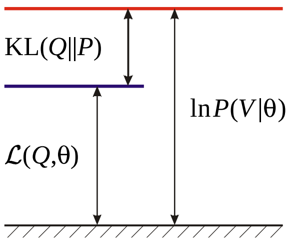
$$\ln P(V|\theta) = \mathcal{L}(Q,\theta) + \mathrm{KL}(Q\|P)$$

where

$$\begin{aligned}\mathcal{L}(Q,\theta) &= \sum_H Q(H|V) \ln \frac{P(H,V|\theta)}{Q(H|V)} \\ \mathrm{KL}(Q\|P) &= -\sum_H Q(H|V) \ln \frac{P(H|V,\theta)}{Q(H|V)}\end{aligned}$$

- Kullback-Leibler divergence satisfies $\mathrm{KL}(Q\|P) \geq 0$ with equality if and only if $Q = P$
- Hence $\mathcal{L}(Q,\theta)$ is a rigorous lower bound on the log marginal likelihood $\ln P(V|\theta)$

EM: Variational Viewpoint (cont'd)



EM: Variational Viewpoint (cont'd)

- If we maximize $\mathcal{L}(Q,\theta)$ with respect to a free-form Q distribution we obtain

$$Q(H|V,\theta) = P(H|V,\theta)$$

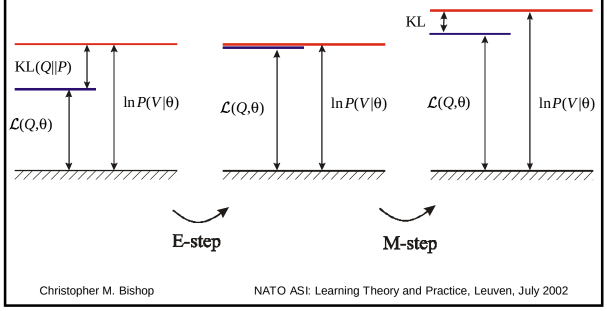
which is the true posterior distribution

- The lower bound then becomes

$$\mathcal{L}(Q,\theta) = \sum_H P(H|V,\theta_{\text{old}}) \ln \frac{P(H,V|\theta)}{P(H|V,\theta_{\text{old}})}$$

which, as a function of θ is the expected complete-data log likelihood (up to an additive constant)

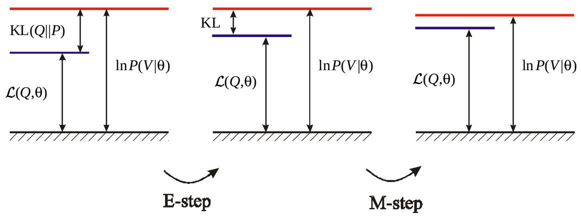
EM: Variational Viewpoint (cont'd)



Generalizations of EM

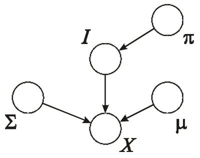
- What if the M-step is intractable?
  - Use non-linear optimization to increase $\mathcal{L}(Q,\theta)$ w.r.t. θ
- What if the E-step is intractable?
  - Perform a partial optimization of $\mathcal{L}(Q,\theta)$ w.r.t. Q
  - e.g. define a parametric family $Q(H|V,\psi)$
  - An alternative approach will be discussed later

Variational EM



Bayesian Learning

- Introduce prior distributions over parameters $P_i(\theta_i)$
- Equivalent to graph with additional hidden variables
- Learning becomes inference on the expanded graph
- No distinction between variables and parameters
- No M-step, just an E-step
- Example: mixture of Gaussians



Variational Inference

- We have already seen that the posterior distribution may be approximated by maximizing a lower bound

$$\mathcal{L}(Q) = \sum_H Q(H|V) \ln \frac{P(H,V)}{Q(H|V)}$$

- For a suitable choice of $Q(H|V)$ this summation may be tractable even though

$$\ln P(V) = \ln \sum_H P(H,V)$$

is not

Variational Inference (cont'd)

- The goal is to choose a sufficiently simple family of distributions $Q(H|V)$ that $\mathcal{L}(Q)$ can be evaluated
- However, the family should also be sufficiently rich that a good approximation to the true posterior can be obtained
- One possibility is to use a parametric family of distributions $Q(H|V,\psi)$

Variational Inference (cont'd)

- Here we consider an alternative approach based on the assumption that $Q(H)$ factorizes with respect to subsets of nodes

$$Q(H) = \prod_i Q_i(H_i)$$

where we leave the conditioning on V implicit

- Substituting into $\mathcal{L}(Q)$ and dissecting out the contribution from $Q_i(H_i)$ we obtain

$$\mathcal{L}(Q) = \sum_{H_i} Q_i(H_i) \sum_{H_{j\neq i}} \prod_{j \neq i} Q_j(H_j) \ln P(H,V)$$
$$- \sum_{H_i} Q_i(H_i) \ln Q_i(H_i) - \sum_{j\neq i} \sum_{H_j} Q_j(H_j) \ln Q_j(H_j)$$

Variational Inference (cont'd)

- We recognize this as the negative KL divergence between $Q_i(H_i)$ and an effective distribution given by

$$\ln P^\star = \sum_{H_{j\neq i}} \prod_{j\neq i} Q_j(H_j) \ln P(H,V)$$

- Hence if we perform a free-form optimization over $Q(H_i)$ we obtain

$$Q_i(H_i) = \frac{\exp \langle \ln P(H,V) \rangle_{k\neq i}}{\sum_{H_i} \exp \langle \ln P(H,V) \rangle_{k\neq i}}$$

Variational Inference (cont'd)

- This is an implicit solution which depends on all $Q_{j\neq i}$
- Hence we initialize the factors $Q_i(H_i)$ and then cyclically update to convergence
- For conjugate-exponential choices of the conditional distributions $P(X_k|\mathrm{pa}_k)$ in the directed graph, the solutions for $Q_i(H_i)$ will have closed form

Overview of Part Two

- Exact inference and the junction tree
- MCMC
- Variational methods and EM
- Example
- General variational inference engine

Examples of Variational Inference

- Hidden Markov models (MacKay)
- Neural networks (Hinton)
- Bayesian PCA (Bishop)
- Independent Component Analysis (Attias)
- Mixtures of Gaussians (Attias; Ghahramani and Beal)
- Mixtures of Bayesian PCA (Bishop and Winn)
- Flexible video sprites (Frey *et al.*)
- Audio-video fusion for tracking (Attias *et al.*)
- Latent Dirichlet Allocation (Jordan *et al.*)
- Relevance Vector Machine (Bishop and Tipping)
- …

Illustration: Relevance Vector Machine (RVM)

- Limitations of the support vector machine (SVM):
  - – Two classes
  - – Large number of kernels (in spite of sparsity)
  - – Kernels must satisfy Mercer criterion
  - – Cross-validation to set parameters C (and ε)
  - – Decisions at outputs instead of probabilities

Illustration: RVM (cont'd)

- The Relevance Vector Machine (Tipping, 1999) is a probabilistic regression or classification model
- Alternative to SVM, *not* a Bayesian interpretation of SVM
- Properties
  - – Comparable error rates to SVM on new data
  - – Comparable training speeds
  - – No cross-validation to set parameters C (and ε)
  - – Applicable to wide choice of basis function
  - – Multi-class classification
  - – Probabilistic outputs
  - – Dramatically fewer kernels
- Original RVM based on type II maximum likelihood – here we consider a variational treatment

Illustration: RVM (cont'd)

- Linear model as for SVM

$$y(\mathbf{x}, \mathbf{w}) = \sum_{m=0}^{M} w_m \phi_m(\mathbf{x}) = \mathbf{w}^{\top} \boldsymbol{\phi}$$

- Input vectors $\{\mathbf{x}_n\}_{n=1}^{N}$ and targets $\{t_n\}_{n=1}^{N}$
- Regression

$$P(t|\mathbf{x}, \mathbf{w}, \tau) = \mathcal{N}(t|y(\mathbf{x}, \mathbf{w}), \tau^{-1})$$

- Likelihood function

$$P(T|X, \mathbf{w}, \tau) = \prod_{n=1}^{N} P(t_n|\mathbf{x}_n, \mathbf{w}, \tau)$$

Illustration: RVM (cont'd)

- Gaussian prior for w_m with hyper-parameters α_m

$$P(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{m=0}^{N} \mathcal{N}(w_m|0, \alpha_m^{-1})$$

- Gamma hyper-priors over τ and α_m

$$\begin{aligned} P(\alpha_m) &= \Gamma(\alpha_m|a, b) \equiv \frac{b^a \alpha_m^{a-1} e^{-b\alpha_m}}{\Gamma(a)} \\ P(\tau) &= \Gamma(\tau|c, d) \end{aligned}$$

Illustration: RVM (cont'd)

- Graphical model representation

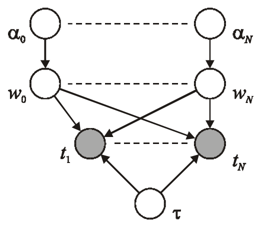


Illustration: RVM (cont'd)

- Variational posterior distribution

$$Q(\mathbf{w}, \boldsymbol{\alpha}, \tau) = Q_{\mathbf{w}}(\mathbf{w}) Q_{\boldsymbol{\alpha}}(\boldsymbol{\alpha}) Q_{\tau}(\tau)$$

- Analytical solutions for optimum factors

$$\begin{aligned} Q_{\mathbf{w}}(\mathbf{w}) &= \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}_{\mathbf{w}}, \boldsymbol{\Sigma}_{\mathbf{w}}) \\ Q_{\tau}(\tau) &= \Gamma(\tau|\tilde{c}, \tilde{d}) \\ Q_{\boldsymbol{\alpha}}(\boldsymbol{\alpha}) &= \prod_{m=0}^{N} \Gamma(\alpha_m|\tilde{a}_m, \tilde{b}_m) \end{aligned}$$

Illustration: RVM (cont'd)

- Sufficient statistics given by

$$\begin{aligned} \boldsymbol{\Sigma}_{\mathbf{w}} &= \left(\mathrm{diag}\langle\alpha_m\rangle + \langle\tau\rangle \sum_{n=1}^{N} \boldsymbol{\phi}_n \boldsymbol{\phi}_n^{\top} \right)^{-1} \\ \boldsymbol{\mu}_{\mathbf{w}} &= \langle\tau\rangle \boldsymbol{\Sigma}_{\mathbf{w}} \sum_{n=1}^{N} \boldsymbol{\phi}_n t_n \\ \tilde{a}_m &= a + 1/2, \qquad \tilde{b}_m = b + \langle w_m^2 \rangle / 2 \\ \tilde{c} &= c + (N+1)/2 \\ \tilde{d} &= d + \frac{1}{2} \sum_{n=1}^{N} t_n^2 - \langle\mathbf{w}\rangle^{\top} \sum_{n=1}^{N} \boldsymbol{\phi}_n t_n \\ &\quad + \frac{1}{2} \sum_{n=1}^{N} \boldsymbol{\phi}_n^{\top} \langle\mathbf{w}\mathbf{w}^{\top}\rangle \boldsymbol{\phi}_n \end{aligned}$$

Illustration: RVM (cont'd)

- Moments given by

$$\begin{aligned} \langle\mathbf{w}\rangle &= \boldsymbol{\mu}_{\mathbf{w}} \\ \langle\mathbf{w}\mathbf{w}^{\top}\rangle &= \boldsymbol{\Sigma}_{\mathbf{w}} + \boldsymbol{\mu}_{\mathbf{w}} \boldsymbol{\mu}_{\mathbf{w}}^{\top} \\ \langle\alpha_m\rangle &= \tilde{a}_m / \tilde{b}_m \\ \langle\ln \alpha_m\rangle &= \psi(\tilde{a}_m) - \ln \tilde{b}_m \\ \langle\tau\rangle &= \tilde{c} / \tilde{d} \\ \langle\ln \tau\rangle &= \psi(\tilde{c}) - \ln \tilde{d} \end{aligned}$$

where the di-gamma function is defined by

$$\psi(a) = \frac{d}{da} \ln \Gamma(a)$$

Illustration: RVM (cont'd)

- A high proportion of α_m are driven to large values in the posterior distribution, giving a *sparse* model
  Lower bound $\mathcal{L}$ can also be evaluated explicitly
- Variational treatment can be extended to classification

Illustration: RVM (cont'd)

- Synthetic data: noisy sinusoid

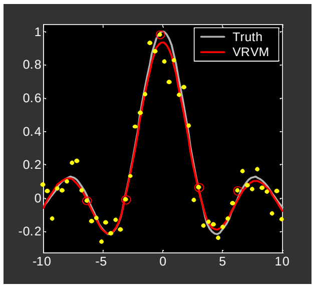
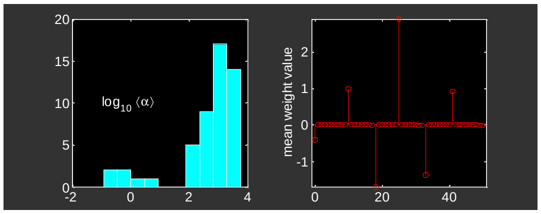


Illustration: RVM (cont'd)

- Synthetic data: noisy sinusoid



Sparsity

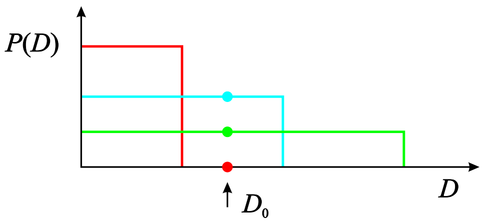


Illustration: RVM (cont’d)

- Classification example
- 100 training points and 150 test points from Ripley's synthetic data set
- Gaussian kernels, with width parameter optimized for the SVM classifier by 5-fold cross-validation

Illustration: RVM (cont’d)

- Results from SVM

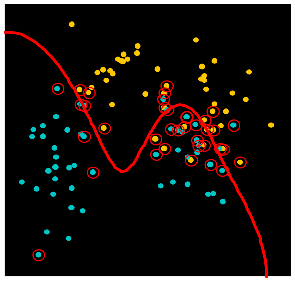


Illustration: RVM (cont’d)

- Results from VRVM

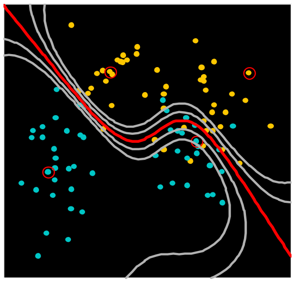


Illustration: RVM (cont’d)

- Summary of results on Ripley data (Bayes error rate 8%)

Model	Error	No. kernels
SVM	10.6%	38
VRVM	9.2%	4

Illustration: RVM (cont’d)

- Additional regression results

	Errors		Kernels	
	SVR	RVR	SVR	RVR
Friedman #1	2.92	2.80	116.6	59.4
Friedman #2	4140	3505	110.3	6.9
Friedman #3	0.0202	0.0164	106.5	11.5
Boston Housing	8.04	7.46	142.8	39.0

Illustration: RVM (cont'd)

- Additional classification results

	Errors			Kernels		
	SVM	GP	RVM	SVM	GP	RVM
Pima Indians	67	68	65	109	200	4
U.S.P.S.	4.4%	-	5.1%	2540	-	316

Overview of Part Two

- Exact inference and the junction tree
- MCMC
- Variational methods and EM
- Example
- General variational inference engine

General Variational Framework

- Currently for each new model we have to derive the variational update equations and then subsequently we write application-specific code to find the solution
- Can we build a general-purpose inference engine which automates these procedures?

VIBES

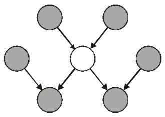
- *Variational Inference for Bayesian Networks*
- Bishop, Spiegelhalter and Winn (1999)
- A general inference engine using variational methods
- VIBES will be made available on the WWW

VIBES (cont'd)

- A key observation is that in the general solution

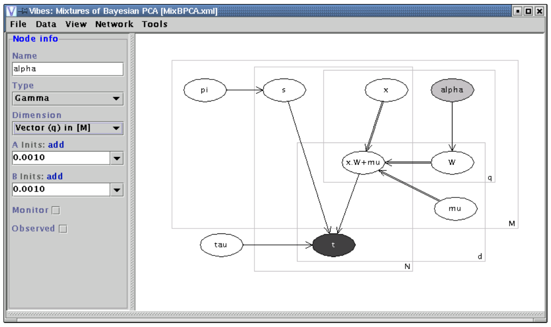
$$Q_i(H_i) = \frac{\exp\langle \ln P(H,V)\rangle_{k\neq i}}{\sum_{H_i}\exp\langle \ln P(H,V)\rangle_{k\neq i}}$$

the update for a particular node (or group of nodes) depends only on other nodes in the *Markov blanket*



- Permits a local object-oriented implementation which is independent of the particular graph structure

VIBES (cont'd)



Vibes: Mixtures of Bayesian PCA [MixBPCA.xml]

Node info

Name: alpha

Type: Gamma

Dimension: Vector (q) in [M]

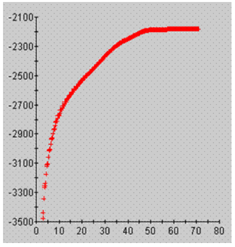
A Inits: add 0.0010

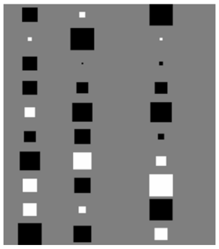
B Inits: add 0.0010

Monitor □

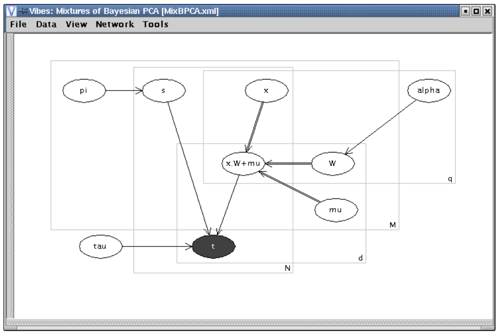
Observed □

VIBES (cont'd)





VIBES (cont'd)



Summary of Part Two

- Exact inference algorithms can be formulated in terms of graphical manipulations (the junction tree)
- Bayesian learning is just inference on an expanded graph
- Variational methods provide a powerful semi-analytical tool for approximate inference in graphical models

Bibliography

- *Probabilistic Networks and Expert Systems.* R. G. Cowell, A. P. Dawid, S. L. Lauritzen and D. J. Spiegelhalter (1999). Springer.
- *Graphical Models*. S. L. Lauritzen. Oxford University Press.
- *Learning in Graphical Models*. M. I. Jordan, Ed (1998). MIT Press.
- *Graphical Models for Machine Learning and Digital Communication*. B. J. Frey (1998). MIT Press.
- Tutorial on Variational Approximation Methods. T. S. Jaakkola, MIT.
- Variational Relevance Vector Machines. C. M. Bishop and M. E. Tipping (2000). In *Proc. 16th Conference on Uncertainty in Artificial Intelligence* (UAI). Ed. C. Boutilier and M. Goldszmidt, Morgan Kaufmann, 46–53.
- Propagation algorithms for variational Bayesian learning. Z. Ghahramani, and M. J. Beal (2001). *Neural Information Processing Systems* **13**, MIT Press.

Viewgraphs and tutorials available from:

research.microsoft.com/~cmbishop