

Readings:

K&F: 11.3, 11.5
Yedidia et al. paper from the class website
Chapter 9 - Jordan

Loopy Belief Propagation Generalized Belief Propagation Unifying Variational and GBP Learning Parameters of MNs

Graphical Models – 10708

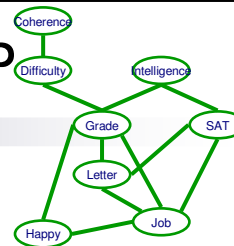
Carlos Guestrin

Carnegie Mellon University

November 10th, 2006

1

More details on Loopy BP



■ Numerical problem:

- messages < 1 get multiplied together as we go around the loops
- numbers can go to zero
- normalize messages to one:

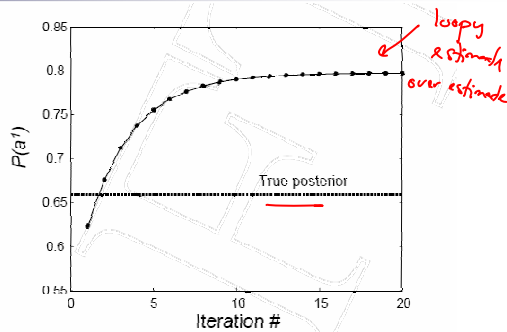
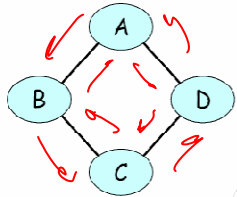
$$\delta_{i \rightarrow j}(X_j) = \frac{1}{Z_{i \rightarrow j}} \sum_{x_i} \phi_i(x_i) \phi_{ij}(x_i, X_j) \prod_{k \in \mathcal{N}(i) - j} \delta_{k \rightarrow i}(x_i)$$

- $Z_{i \rightarrow j}$ doesn't depend on X_j , so doesn't change the answer

■ Computing node “beliefs” (estimates of probs.):

$$\hat{P}(X_i) = \frac{1}{Z_i} \phi_i(X_i) \prod_{k \in \mathcal{N}(i)} \delta_{k \rightarrow i}(X_i)$$

An example of running loopy BP



usually over confident ...

(Non-)Convergence of Loopy BP

Loopy BP can oscillate!!!

- oscillations can be small
- oscillations can be really bad!

Typically,

- if factors are closer to uniform, loopy does well (converges)
- if factors are closer to deterministic, loopy doesn't behave well

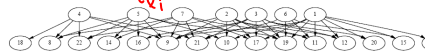
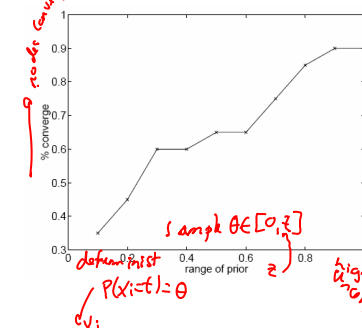
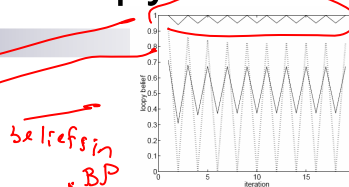
One approach to help: damping messages

- new message is average of old message and new one:

$$\tilde{\sigma}_{i \rightarrow j} = \lambda \tilde{\sigma}_{i \rightarrow j}^{\text{last iteration}} + (1-\lambda) \tilde{\sigma}_{i \rightarrow j}^{\text{computed}}$$

- often better convergence

but, when damping is required to get convergence, result often bad



graphs from Murphy et al. '99

Loopy BP in Factor graphs

- What if we don't have pairwise Markov nets?

1. Transform to a pairwise MN
2. Use Loopy BP on a factor graph



- Message example:
 - from node to factor:
 - from factor to node:

10.708 - ©Carlos Guestrin 2006

5

Loopy BP in Factor graphs

- From node i to factor j :
 - $F(i)$ factors whose scope includes X_i

$$\delta_{i \rightarrow j}(X_i) \propto \prod_{k \in \mathcal{F}(i) - j} \delta_{k \rightarrow i}(X_i)$$



- From factor j to node i :
 - Scope $[\phi_j] = \mathbf{Y} \cup \{X_i\}$

$$\delta_{j \rightarrow i}(X_i) \propto \sum_{\mathbf{y}} \phi_j(X_i, \mathbf{y}) \prod_{X_k \in \text{Scope}[\phi_j] - X_i} \delta_{k \rightarrow j}(x_k)$$

10.708 - ©Carlos Guestrin 2006

6

What you need to know about loopy BP

- Application of belief propagation in loopy graphs
- Doesn't always converge
 - damping can help
 - good message schedules can help (see book)
- If converges, often to incorrect, but useful results
- Generalizes from pairwise Markov networks by using factor graphs

10.708 – ©Carlos Guestrin 2006

7

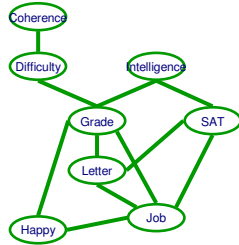
Announcements

- Monday's special recitation
 - Pradeep Ravikumar on exciting new approximate inference algorithms

10.708 – ©Carlos Guestrin 2006

8

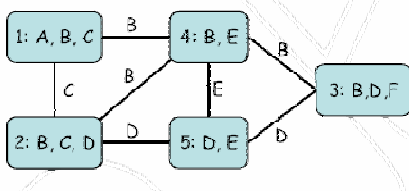
Loopy BP v. Clique trees: Two ends of a spectrum



10.708 – ©Carlos Guestrin 2006

9

Generalize cluster graph



Generalized cluster graph:

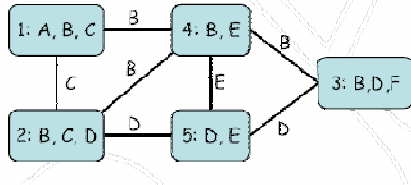
For set of factors F

- Undirected graph
- Each node i associated with a cluster C_i
- *Family preserving*: for each factor $f_j \in F$, \exists node i such that $\text{scope}[f_j] \subseteq C_i$
- Each edge $i - j$ is associated with a set of variables $S_{ij} \subseteq C_i \cap C_j$

10.708 – ©Carlos Guestrin 2006

10

Running intersection property



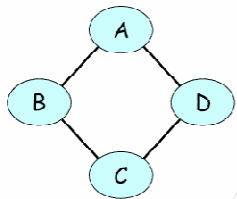
■ (Generalized) Running intersection property (RIP)

- Cluster graph satisfies RIP if whenever $X \in C_i$ and $X \in C_j$, then \exists one and only one path from C_i to C_j where $X \in S_{uv}$ for every edge (u,v) in the path

10.708 – ©Carlos Guestrin 2006

11

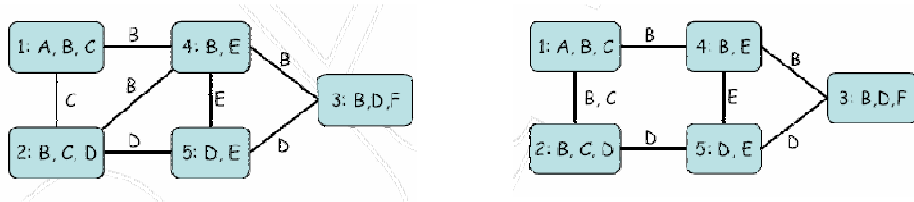
Examples of cluster graphs



10.708 – ©Carlos Guestrin 2006

12

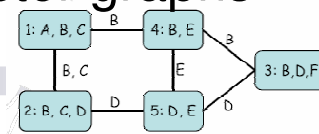
Two cluster graph satisfying RIP with different edge sets



10.708 - ©Carlos Guestrin 2006

13

Generalized BP on cluster graphs satisfying RIP



Initialization:

- Assign each factor ϕ to a clique $\alpha(\phi)$, $\text{Scope}[\phi] \subseteq \mathbf{C}_{\alpha(\phi)}$
- Initialize cliques: $\pi_i^0(C_i) \propto \prod_{\phi: \alpha(\phi)=i} \phi$
- Initialize messages: $\delta_{j \rightarrow i} = 1$

While not converged, send messages:

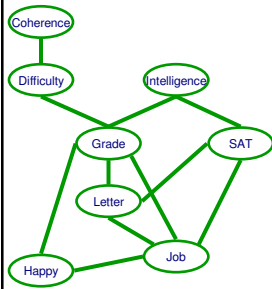
$$\delta_{i \rightarrow j}(S_{ij}) \propto \sum_{C_i - S_{ij}} \pi_i^0(C_i) \prod_{k \in \mathcal{N}(i) - j} \delta_{k \rightarrow i}(S_{ik})$$

Belief:

10.708 - ©Carlos Guestrin 2006

14

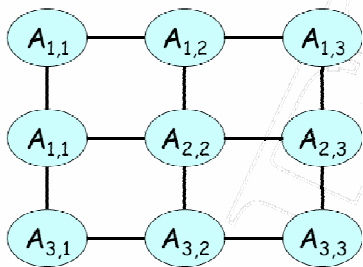
Cluster graph for Loopy BP



10.708 - ©Carlos Guestrin 2006

15

What if the cluster graph doesn't satisfy RIP



10.708 - ©Carlos Guestrin 2006

16

Region graphs to the rescue

- Can address generalized cluster graphs that don't satisfy RIP using *region graphs*:
 - Yedidia et al. from class website
- Example in your homework! 😊
- Hint – From Yedidia et al.:
 - Section 7 – defines region graphs
 - Section 9 – message passing on region graphs
 - Section 10 – An example that will help you a lot!!! 😊

10.708 – ©Carlos Guestrin 2006

17

Revisiting Mean-Fields

$$\ln Z = F[P_{\mathcal{F}}, Q] + D(Q||P_{\mathcal{F}}) \quad F[P_{\mathcal{F}}, Q] = \sum_{\phi \in \mathcal{F}} E_Q[\ln \phi] + H_Q(\mathcal{X})$$

- Choice of Q:
- Optimization problem:

$$\max_Q F[P_{\mathcal{F}}, Q] = \sum_{\phi \in \mathcal{F}} E_Q[\ln \phi] + \sum_j H_{Q_j}(X_j), \quad \forall i, \sum_{x_i} Q_i(x_i) = 1$$

10.708 – ©Carlos Guestrin 2006

18

Interpretation of energy functional

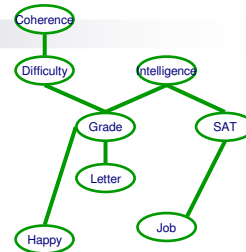
- Energy functional:
$$F[P_{\mathcal{F}}, Q] = \sum_{\phi \in \mathcal{F}} E_Q[\ln \phi] + H_Q(\mathcal{X})$$
- Exact if P=Q:
$$\ln Z = F[P_{\mathcal{F}}, Q] + D(Q||P_{\mathcal{F}})$$
- View problem as an approximation of entropy term:

10.708 - ©Carlos Guestrin 2006

19

Entropy of a tree distribution

- Entropy term:
- Joint distribution:
- Decomposing entropy term:



- More generally:
$$H_P(\mathbf{X}) = \sum_{(i,j) \in E} H(X_i, X_j) - \sum_i (d_i - 1) H(X_i)$$
 - d_i number neighbors of X_i

10.708 - ©Carlos Guestrin 2006

20

Loopy BP & Bethe approximation

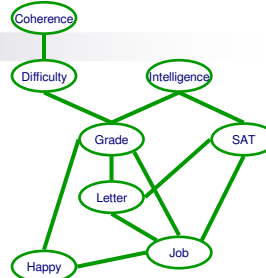
Energy functional: $F[P_{\mathcal{F}}, Q] = \sum_{\phi \in \mathcal{F}} E_Q[\ln \phi] + H_Q(\mathcal{X})$

Bethe approximation of Free Energy:

□ use entropy for trees, but loopy graphs:

$$\tilde{F}[P_{\mathcal{F}}, Q] = \sum_{(i,j) \in E} E_{\pi_{ij}}[\ln \phi_{ij}] + \sum_{(i,j) \in E} H_{\pi_{ij}}(X_i, X_j) - \sum_i (d_i - 1) H_{\pi_i}(X_i)$$

Theorem: If Loopy BP converges, resulting π_{ij} & π_i are stationary point (usually local maxima) of Bethe Free energy!



GBP & Kikuchi approximation

Exact Free energy: Junction Tree

$$F[P_{\mathcal{F}}, Q] = \sum_{(i,j) \in E} E_{\pi_{ij}}[\ln \phi_{ij}] + \sum_i H_{\pi_{C_i}}(C_i) - \sum_{(i,j) \in \mathcal{T}} H_{\pi_{S_{ij}}}(S_{ij})$$

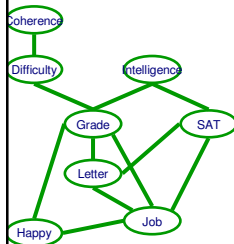
Bethe Free energy:

$$\tilde{F}[P_{\mathcal{F}}, Q] = \sum_{(i,j) \in E} E_{\pi_{ij}}[\ln \phi_{ij}] + \sum_{(i,j) \in E} H_{\pi_{ij}}(X_i, X_j) - \sum_i (d_i - 1) H_{\pi_i}(X_i)$$

Kikuchi approximation: Generalized cluster graph

- spectrum from Bethe to exact
- entropy terms weighted by counting numbers
- see Yedidia et al.

Theorem: If GBP converges, resulting π_{C_i} are stationary point (usually local maxima) of Kikuchi Free energy!



What you need to know about GBP

- Spectrum between Loopy BP & Junction Trees:
 - More computation, but typically better answers
- If satisfies RIP, equations are very simple
- General setting, slightly trickier equations, but not hard
- Relates to variational methods: Corresponds to local optima of approximate version of energy functional

10.708 - ©Carlos Guestrin 2006

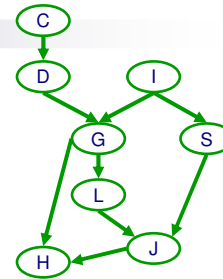
23

Learning Parameters of a BN

- Log likelihood decomposes:

$$\ell(\mathcal{D} : \theta) = \log P(\mathcal{D} | \theta) = m \sum_i \sum_{\mathbf{Pa}_{x_i}} \hat{P}(x_i, \mathbf{Pa}_{x_i}) \log P(x_i | \mathbf{Pa}_{x_i})$$

- Learn each CPT independently
- Use counts



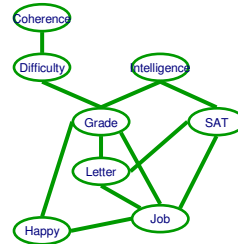
$$\hat{P}(\mathbf{u}) = \frac{\text{Count}(\mathbf{U} = \mathbf{u})}{m}$$

10.708 - ©Carlos Guestrin 2006

24

Log Likelihood for MN

- Log likelihood of the data:



10.708 – ©Carlos Guestrin 2006

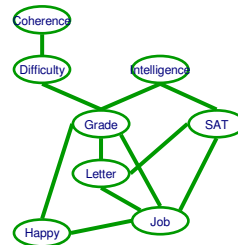
25

Log Likelihood doesn't decompose for MNs

$$\hat{P}(\mathbf{u}) = \frac{\text{Count}(\mathbf{U} = \mathbf{u})}{m}$$

- Log likelihood:

$$\ell(\mathcal{D} : \theta) = \log P(\mathcal{D} | \theta, \mathcal{G}) = m \sum_i \sum_{\mathbf{c}_i} \hat{P}(\mathbf{c}_i) \log \psi_i(\mathbf{c}_i) - m \log Z$$



- A convex problem
 - Can find global optimum!!
- Term $\log Z$ doesn't decompose!!

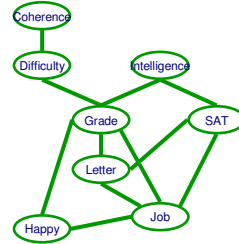
10.708 – ©Carlos Guestrin 2006

26

Derivative of Log Likelihood for MNs

$$\hat{P}(\mathbf{u}) = \frac{\text{Count}(\mathbf{U} = \mathbf{u})}{m}$$

$$\ell(\mathcal{D} : \theta) = \log P(\mathcal{D} | \theta, \mathcal{G}) = m \sum_i \sum_{\mathbf{c}_i} \hat{P}(\mathbf{c}_i) \log \psi_i(\mathbf{c}_i) - m \log Z$$



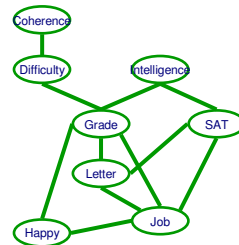
Derivative of Log Likelihood for MNs

$$\hat{P}(\mathbf{u}) = \frac{\text{Count}(\mathbf{U} = \mathbf{u})}{m}$$

$$\ell(\mathcal{D} : \theta) = \log P(\mathcal{D} | \theta, \mathcal{G}) = m \sum_i \sum_{\mathbf{c}_i} \hat{P}(\mathbf{c}_i) \log \psi_i(\mathbf{c}_i) - m \log Z$$

- Derivative:

$$\frac{\partial \ell}{\partial \psi_i(\mathbf{c}_i)} = \frac{m \hat{P}(\mathbf{c}_i)}{\psi_i(\mathbf{c}_i)} - \frac{m P_{\mathcal{F}}(\mathbf{c}_i)}{\psi_i(\mathbf{c}_i)}$$



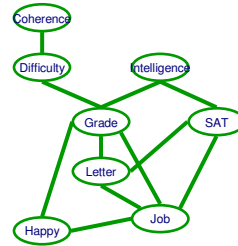
- Setting derivative to zero
- Can optimize using gradient ascent
 - Let's look at a simpler solution

Iterative Proportional Fitting (IPF)

$$\hat{P}(u) = \frac{\text{Count}(U = u)}{m}$$

$$\frac{\partial \ell}{\partial \psi_i(c_i)} = \frac{m\hat{P}(c_i)}{\psi_i(c_i)} - \frac{mP_{\mathcal{F}}(c_i)}{\psi_i(c_i)}$$

- Setting derivative to zero:
- Fixed point equation:
- Iterate and converge to optimal parameters
 - Each iteration, must compute:



10.708 - ©Carlos Guestrin 2006

29

What you need to know about learning MN parameters?

- BN parameter learning easy
- MN parameter learning doesn't decompose!
- Learning requires inference!
- Apply gradient ascent or IPF iterations to obtain optimal parameters

10.708 - ©Carlos Guestrin 2006

30