CS839: Probabilistic Graphical Models

Lecture 11: Loopy Belief Propagation

Theo Rekatsinas
Back to inference

• Compute the likelihood of observed data
• Compute the marginal distribution $p(X)$ over a particular subset of nodes $X$
• Compute the conditional distribution $p(X|Y)$ for disjoint subsets A and B
• Compute the mode of the density $\hat{x} = \arg\max_{x \in X^m} p(x)$
• Methods we saw:

**Message Passing**
- Brute force
- Elimination

(Forward-backward, Max-product /BP, Junction Tree)

Individual computations independent
Sharing intermediate terms
Recall: Sum-Product

- Tree-structure graphical models

\[ p(x_1, \cdots, x_m) = \frac{1}{Z} \prod_{s \in V} \psi_s(x_s) \prod_{(s, t) \in E} \psi_{st}(x_s, x_t) \]

- Message passing on trees:

\[ M_{t \rightarrow s}(x_s) \leftarrow \kappa \sum_{x_t'} \left\{ \psi_{st}(x_s, x_t') \psi_t(x_t') \prod_{u \in N(t) \setminus s} M_{u \rightarrow t}(x_t') \right\} \]

- For trees, mp converges to a unique fixed point after a finite number of iterations
Recall: Junction Tree

• General algorithm on graphs with cycles

• Message passing on clique trees:

\[ \tilde{\phi}_S(x_S) \leftarrow \sum_{x_{B \setminus S}} \phi_B(x_B) \]

\[ \phi_C(x_C) \leftarrow \frac{\tilde{\phi}_S(x_S)}{\phi_S(x_S)} \phi_C(x_C) \]
Recall: Local Consistency

• Given a set of functions \( \{ \tau_c \} \) for cliques and \( \{ \tau_s \} \) for separators
• They are locally consistent if:

\[
\sum_{x'_S} \tau_S(x'_S) = 1, \quad \forall S \in S
\]

\[
\sum_{x'_C | x'_S = x_S} \tau_C(x'_C) = \tau_S(x_S), \quad \forall C \in C, \ S \subset C
\]

• For junction trees, local consistency is equivalent to global consistency.
Why approximate inference?

- Why can’t we just use the junction tree algorithm on this graph?

\[ p(X) = \frac{1}{Z} \exp \left\{ \sum_{i<j} \theta_{ij} X_i X_j + \sum_i \theta_{i0} X_i \right\} \]
Approaches to inference

• Exact inference algorithms
  • The elimination algorithm
  • Message-passing algorithm (sum-product, belief propagation)
  • Junction tree algorithm

• Approximate inference techniques
  • Variational algorithms
    • Loopy belief propagation
    • Mean field approximation
  • Stochastic simulation / sampling methods
  • Markov chain Monte Carlo methods
Approaches to inference

• Exact inference algorithms
  • The elimination algorithm
  • Message-passing algorithm (sum-product, belief propagation)
  • Junction tree algorithm

• Approximate inference techniques
  • Variational algorithms
    • Loopy belief propagation (TODAY)
    • Mean field approximation
  • Stochastic simulation / sampling methods
  • Markov chain Monte Carlo methods
Recap: Belief Propagation

• BP message-update rules

\[ M_{i\rightarrow j}(x_j) \propto \sum_{x_i} \psi_j(x_i, x_j) \psi_j(x_i) \prod_k M_{k\rightarrow i}(x_i) \]

\[ b_i(x_i) \propto \psi_i(x_i) \prod_k M_k(x_k) \]

• BP on trees always converges to exact marginals
Factor Graphs

- Useful to look explicitly at the messages being passed
  - Messages from variable to factors
  - Messages from factors to variables
Beliefs and Messages in FGs

\[ b_i(x_i) \propto f_i(x_i) \prod_{a \in N(i)} m_{a \rightarrow i}(x_i) \]

\[ m_{i \rightarrow a}(x_i) = \prod_{c \in N(i) \setminus a} m_{c \rightarrow i}(x_i) \]

\[ b_a(X_a) \propto f_a(X_a) \prod_{i \in N(a)} m_{i \rightarrow a}(x_i) \]

\[ m_{a \rightarrow i}(x_i) = \sum_{X_a \setminus x_i} f_a(X_a) \prod_{j \in N(a) \setminus i} m_{j \rightarrow a}(x_j) \]
Belief Propagation on loopy graphs

- BP message-update rules

\[ M_{i \rightarrow j}(x_j) \propto \sum_{x_i} \psi_j(x_i, x_j) \psi_i(x_i) \prod_k M_{k \rightarrow i}(x_i) \]

- May not converge or converge to a wrong solution

\[ b_i(x_i) \propto \psi_i(x_i) \prod_k M_k(x_k) \]
Loopy Belief Propagation

• A fixed point iteration procedure that tries to minimize a free energy function

• Start with random initialization of messages and beliefs
  • Repeat until convergence

\[
b_i(x_i) \propto \prod_{a \in N(i)} m_{a \rightarrow i}(x_i) \quad b_a(X_a) \propto f_a(X_a) \prod_{i \in N(a)} m_{i \rightarrow a}(x_i)
\]

\[
m_{i \rightarrow a}^{new}(x_i) = \prod_{c \in N(i) \setminus a} m_{c \rightarrow i}(x_i) \quad m_{a \rightarrow i}^{new}(x_i) = \sum_{X_a \setminus x_i} f_a(X_a) \prod_{j \in N(a) \setminus i} m_{j \rightarrow a}(x_j)
\]

• At convergence, stationarity properties are guaranteed
• However, not guaranteed to converge
Loopy Belief Propagation

• If BP is used on graphs with loops, messages may circulate indefinitely

• Still let’s run it!

• Empirically, a good approximation is still achievable
  • Stop after fixed # of iterations
  • Stop when there is no significant change in beliefs
  • If solution is not oscillatory but converges, it usually is a good approximation
Approximate Inference

• Consider the actual distribution \( P \)

\[
P(X) = \frac{1}{Z} \prod_{f_a \in F} f_a(X_a)
\]

• We wish to find a distribution \( Q \) such that \( Q \) is a good approximation of \( P \)

• Let’s take the definition of KL-divergence to specify “good”

\[
KL(Q_1 \parallel Q_2) = \sum_X Q_1(X) \log \left( \frac{Q_1(X)}{Q_2(X)} \right)
\]

• \( KL(Q_1 \parallel Q_2) \geq 0 \)
• \( KL(Q_1 \parallel Q_2) = 0 \) iff \( Q_1 = Q_2 \)
• \( KL(Q_1 \parallel Q_2) \neq KL(Q_2 \parallel Q_1) \)
Which direction shall we take?

• Computing $\text{KL}(P \mid \mid Q)$ requires inference
• But $\text{KL}(Q \mid \mid P)$ can be computed without performing inference on $P$

$$KL(Q \mid \mid P) = \sum_x Q(X) \log \left( \frac{Q(X)}{P(X)} \right)$$
$$= \sum_x Q(X) \log Q(X) - \sum_x Q(X) \log P(X)$$
$$= -H_Q(X) - E_Q \log P(X)$$

$$P(X) = \frac{1}{Z} \prod_{f_a \in F} f_a(X_a)$$

$$KL(Q \mid \mid P) = -H_Q(X) - E_Q \log \left( \frac{1}{Z} \prod_{f_a \in F} f_a(X_a) \right)$$
$$= -H_Q(X) - \log \frac{1}{Z} - \sum E_Q \log f_a(X_a)$$
Optimization function

\[
KL(Q \parallel P) = -H_Q(X) - \sum_{f_a \in F} E_Q \log f_a(X_a) + \log Z
\]

\[
F(P, Q)
\]

• We will call \( F(P, Q) \) the “Free Energy”
• What is \( F(P, P) = ? \)
• We also have \( F(P, Q) \geq F(P, P) \)
The Free Energy

$$F(P, Q) = -H_Q(X) - \sum_{f_a \in F} E_Q \log f_a(X_a)$$

- $\sum_{f_a \in F} E_Q \log f_a(X_a)$ can be computed if we have marginals over each $f_a$

- $H_Q = -\sum_X Q(X) \log Q(X)$ is harder. Requires summation over all values

- Computing $F$ is hard in general

- Let’s approximate it with an easy to compute function
Tree Energy Functionals

• Consider the following PGM

• We have \[ b(x) = \prod_a b_a(x_a) \prod_i b_i(x_i)^{1-d_i} \]

\[
H_{\text{tree}} = -\sum_a \sum_{x_a} b_a(x_a) \ln b_a(x_a) + \sum_i (d_i - 1) \sum_{x_i} b_i(x_i) \ln b_i(x_i)
\]

\[
F_{\text{Tree}} = \sum_a \sum_{x_a} b_a(x_a) \ln \frac{b_a(x_a)}{f_a(x_a)} + \sum_i (1-d_i) \sum_{x_i} b_i(x_i) \ln b_i(x_i)
\]

\[= F_{12} + F_{23} + \ldots + F_{67} + F_{78} - F_1 - F_5 - F_2 - F_6 - F_3 - F_7 \]

• We are summing over edges and vertices and is therefore easy to compute
Bethe Approximation to Gibbs Free Energy

• For a general graph, we choose that easy function to be $F_{\text{Bethe}}$

\[
H_{\text{Bethe}} = - \sum_a \sum_{x_a} b_a(x_a) \ln b_a(x_a) + \sum_i (d_i - 1) \sum_{x_i} b_i(x_i)
\]

\[
F_{\text{Bethe}} = - \sum_a \sum_{x_a} b_a(x_a) \ln \frac{b_a(x_a)}{f_a(x_a)} + \sum_i (1 - d_i) \sum_{x_i} b_i(x_i) = - \langle f_a(x_a) \rangle - H_{\text{Bethe}}
\]

\[
F_{\text{Bethe}} = F_{12} + F_{23} + \cdots + F_{67} + F_{78} - F_1 - F_5 - 2F_2 - 2F_6 - \cdots - F_8
\]

Equal to exact Gibbs free energy when the factor graph is a tree
In general $H_{\text{Bethe}}$ is not the same as the $H$ of a tree
Bethe Approximation

• Pros: Easy to compute, since entropy term involves sum over pairwise and single variables

• Cons:
  • $F_{\text{Bethe}}$ may or may not be well connected to $F(P,Q)$
  • It could be greater, equal or less than $F(P,Q)$

• Optimize each $b(x_a)$
  • For discrete beliefs, constrained opt. with lagrangian multiplier
  • Not always converge
Bethe Free Energy for FGs

\[ H_{\text{Bethe}} = - \sum_a \sum_{x_a} b_a(x_a) \ln b_a(x_a) + \sum_i (d_i - 1) \sum_{x_i} b_i(x_i) \]

\[ F_{\text{Bethe}} = - \sum_a \sum_{x_a} b_a(x_a) \ln \frac{b_a(x_a)}{f_a(x_a)} + \sum_i (1 - d_i) \sum_{x_i} b_i(x_i) = -\langle f_a(x_a) \rangle - H_{\text{Bethe}} \]
Minimizing the Bethe Free Energy

\[ L = F_{\text{Bethe}} + \sum_i \gamma_i \left\{ 1 - \sum_{x_i} b_i(x_i) \right\} \]

\[ + \sum_a \sum_{i \in N(a)} \sum_{x_i} \lambda_{ai}(x_i) \left\{ b_i(x_i) - \sum_{X_a \setminus x_i} b_a(X_a) \right\} \]

Set the derivative to zero
Minimizing the Bethe Free Energy

\[ L = F_{\text{Bethe}} + \sum_i \gamma_i \left\{ \sum_{x_i} b_i(x_i) - 1 \right\} \]

\[ + \sum_{a} \sum_{i \in N(a)} \sum_{x_i} \lambda_{ai}(x_i) \left\{ \sum_{X_a \setminus x_i} b_a(X_a) - b_i(x_i) \right\} \]

\[ \frac{\partial L}{\partial b_i(x_i)} = 0 \quad \iff \quad b_i(x_i) \propto \exp \left( \frac{1}{d_i - 1} \sum_{a \in N(i)} \lambda_{ai}(x_i) \right) \]

\[ \frac{\partial L}{\partial b_a(X_a)} = 0 \quad \iff \quad b_a(X_a) \propto \exp \left( -E_a(X_a) + \sum_{i \in N(a)} \lambda_{ai}(x_i) \right) \]
Bethe = BP on FG

• We have:

\[ b_i(x_i) \propto \exp \left( \frac{1}{d_i - 1} \sum_{a \in N(i)} \lambda_{ai}(x_i) \right) \quad b_a(X_a) \propto \exp \left( -E_a(X_a) + \sum_{i \in N(a)} \lambda_{ai}(x_i) \right) \]

• Set

\[ \lambda_{ai}(x_i) = \log(m_{i \to a}(x_i)) = \log \prod_{b \in N(i) \neq a} m_{b \to i}(x_i) \]

• We get the BP equations

\[ b_i(x_i) \propto f_i(x_i) \prod_{a \in N(i)} m_{a \to i}(x_i) \]

\[ b_a(X_a) \propto f_a(X_a) \prod_{i \in N(a)} m_{i \to a}(x_i) \]
BP Message-update Rules

Using \( b_{a \rightarrow i}(x_i) = \sum_{X_a \setminus x_i} b_a(X_a) \), we get

\[
m_{a \rightarrow i}(x_i) = \sum_{X_a \setminus x_i} f_a(X_a) \prod_{j \in N(a) \setminus i} \prod_{b \in N(j) \setminus a} m_{b \rightarrow j}(x_j)
\]

( A sum product algorithm )
What did we see so far

\[ P(X) = \frac{1}{Z} \prod_{f \in F} f_a(X_a) \]

\[ F(P, Q) = -H_Q(X) - \sum_{f_a \in F} E_{Q} \log f_a(X_a) \]

\[ \hat{F}(P, Q) = \sum_a \sum_{x_a} b_a(x_a) \log \frac{f_a(x_a)}{b_a(x_a)} + \sum_i (1 - d_i) \sum_{x_i} b_i(x_i) \log b_i(x_i) \]

\[ b_i(x_i) \propto \exp \left( \frac{1}{d_i - 1} \sum_{a \in N(i)} \lambda_{ai}(x_i) \right) \]

\[ b_a(X_a) \propto \exp \left( -E_a(X_a) + \sum_{i \in N(a)} \lambda_{ai}(x_i) \right) \]
Loopy BP

• For a distribution $p(X|\theta)$ associated with a complex graph, computing the marginal (or conditional) probability of arbitrary random variables is intractable

• Variational methods
  • Formulating probabilistic inference as an optimization problem

$$q^* = \arg \min_{q \in S} \left\{ F_{\text{Bethe}}(p, q) \right\}$$

$$F_{\text{Bethe}} = \sum_a \sum_{x_a} b_a(x_a) \ln \frac{b_a(x_a)}{f_a(x_a)} + \sum_i (1 - d_i) \sum_{x_i} b_i(x_i) \ln b_i(x_i) = -\langle f_a(x_a) \rangle - H_{\text{Bethe}}$$

$q$: a (tractable) probability distribution
Loopy BP

• But we do not optimize \( q(\mathbf{X}) \) explicitly, we focus on the set of beliefs

\[
b = \{ b_{i,j} = \tau(x_i, x_j), \ b_i = \tau(x_i) \}
\]

• Relax the optimization problem
  • Approximate objective
    \[
    H_q \approx F(b) \quad \quad \quad H_{\text{Bethe}}
    \]
  • Relaxed feasible set
    \[
    \mathcal{M} \rightarrow \mathcal{M}_o \quad (\mathcal{M}_o \supseteq \mathcal{M}) \quad \quad \quad \mathcal{M}_o = \{ \tau \geq 0 \mid \sum_{x_i} \tau(x_i) = 1, \sum_{x_i} \tau(x_i, x_j) = \tau(x_j) \}
    \]

\[
b^* = \arg \min_{b \in \mathcal{M}_o} \{ \langle E \rangle_b + F(b) \}
\]

• Loopy BP: fixed point iteration procedure that tries to solve \( b^* \)
Generalized Belief Propagation

• Belief in a region is the product of:
  • Local information (factors in region)
  • Messages from parent regions
  • Messages into descendant regions from parents who are not descendants

• Message-update rules are obtained by enforcing marginalization constraints (consistency)
Region-based Approximation to the Gibbs Free Energy

Exact: $G[q(X)]$ (intractable)

Regions: $G[\{b_r(X_r)\}]$
Summary

• We defined an objective function $F$ for approximate inference
• We found that optimizing this function was hard
• We first approximated the objective function $F$ with the simpler $F_{\text{Bethe}}$
  • Minima of $F_{\text{Bethe}}$ are fixed points of BP
• Then we saw more complex approximations
  • Generalized Belief Propagation