Variational Methods, Belief Propagation, & Graphical Models

Erik Sudderth
Massachusetts Institute of Technology
Undirected Graphical Models

An undirected graph $\mathcal{G}$ is defined by
\[
\mathcal{V} \rightarrow \text{set of } N \text{ nodes } \{1, 2, \ldots, N\}
\]
\[
\mathcal{E} \rightarrow \text{set of edges } (s, t) \text{ connecting nodes } s, t \in \mathcal{V}
\]
Nodes $s \in \mathcal{V}$ are associated with random variables $x_s$

\[
p(x_A, x_C|x_B) = p(x_A|x_B)p(x_C|x_B)
\]
Nearest-Neighbor Grids

Low Level Vision
- Image denoising
- Stereo
- Optical flow
- Shape from shading
- Superresolution
- Segmentation

\[ x_s \rightarrow \text{unobserved or hidden variable} \]
\[ y_s \rightarrow \text{local observation of } x_s \]
Hidden Markov Models (HMMs)

Visual Tracking

\[ p(x, y) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1}) p(y_t \mid x_t) \]

“Conditioned on the present, the past and future are statistically independent”
Other Graphical Models

Articulated Models

Pictorial Structures (Constellation Models)

Images removed due to copyright considerations.
Outline

**Inference in Graphical Models**
- Pairwise Markov random fields
- Belief propagation for trees

**Variational Methods**
- Mean field
- Bethe approximation & BP

**Extensions of Belief Propagation**
- Efficient message passing implementation
- Generalized BP
- Particle filters and nonparametric BP
Pairwise Markov Random Fields

\[ p(x \mid y) = \frac{1}{Z} \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \prod_{s \in \mathcal{V}} \psi_s(x_s, y) \]

\( \mathcal{V} \rightarrow \) set of \( N \) nodes \( \{1, 2, \ldots, N\} \)

\( \mathcal{E} \rightarrow \) set of edges \((s, t)\) connecting nodes \( s, t \in \mathcal{V} \)

\( Z \rightarrow \) normalization constant (partition function)

- Product of arbitrary positive clique potential functions
- Guaranteed Markov with respect to corresponding graph
Markov Chain Factorizations

\[ p(x \mid y) = \frac{1}{Z} \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \prod_{s \in \mathcal{V}} \psi_s(x_s, y) \]
Energy Functions

\[ p(x \mid y) = \frac{1}{Z} \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \prod_{s \in \mathcal{V}} \psi_s(x_s, y) \]

\[ = \frac{1}{Z} \exp \left\{ - \sum_{(s,t) \in \mathcal{E}} \phi_{st}(x_s, x_t) - \sum_{s \in \mathcal{V}} \phi_s(x_s, y) \right\} \]

\[ = \frac{1}{Z} \exp \{ - \mathcal{E}(x) \} \]

\[ \phi_{st}(x_s, x_t) = - \log \psi_{st}(x_s, x_t) \quad \phi_s(x_s) = - \log \psi_s(x_s) \]

- Interpretation inspired by statistical physics
- Justifications from probability (notational convenience)
Probabilistic Inference

\[ p(x \mid y) = \frac{1}{Z} \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \prod_{s \in \mathcal{V}} \psi_s(x_s, y) \]

Maximum a Posteriori (MAP) Estimate

\[ \hat{x} = \arg \max_x p(x \mid y) \]

Posterior Marginal Densities

\[ p_t(x_t \mid y) = \sum_{x_{\mathcal{V} \setminus t}} p(x \mid y) \]

- Bayes least squares estimate
- Maximizer of the Posterior Marginals (MPM)
- Measures of confidence in these estimates
Inference via the Distributed Law

\[
p_1(x_1) = \sum_{x_2, x_3, x_4} \psi_1(x_1)\psi_{12}(x_1, x_2)\psi_2(x_2)\psi_{23}(x_2, x_3)\psi_3(x_3)\psi_{24}(x_2, x_4)\psi_4(x_4)
\]

\[
= \psi_1(x_1) \sum_{x_2, x_3, x_4} \psi_{12}(x_1, x_2)\psi_2(x_2)\psi_{23}(x_2, x_3)\psi_3(x_3)\psi_{24}(x_2, x_4)\psi_4(x_4)
\]

\[
= \psi_1(x_1) \sum_{x_2} \psi_{12}(x_1, x_2)\psi_2(x_2) \sum_{x_3, x_4} \psi_{23}(x_2, x_3)\psi_3(x_3)\psi_{24}(x_2, x_4)\psi_4(x_4)
\]

\[
= \psi_1(x_1) \sum_{x_2} \psi_{12}(x_1, x_2)\psi_2(x_2) \left[ \sum_{x_3} \psi_{23}(x_2, x_3)\psi_3(x_3) \right] \cdot \left[ \sum_{x_4} \psi_{24}(x_2, x_4)\psi_4(x_4) \right]
\]

\[
m_{21}(x_1) = \sum_{x_2} \psi_{12}(x_1, x_2)\psi_2(x_2)m_{32}(x_2)m_{42}(x_2)
\]

\[
m_{32}(x_2)
\]

\[
m_{42}(x_2)
\]
Belief Propagation (Sum-Product)

**BELIEFS:** Posterior marginals (possibly approximate)

\[
q_t(x_t \mid y) = \alpha \psi_t(x_t, y) \prod_{u \in \Gamma(t)} m_{ut}(x_t)
\]

\(\Gamma(t)\) \rightarrow \text{neighborhood of node } t \\
(adjacent nodes)

**MESSAGES:** Sufficient statistics (possibly approximate)

\[
m_{ts}(x_s) = \alpha \sum_{x_t} \psi_{st}(x_s, x_t) \psi_t(x_t, y) \prod_{u \in \Gamma(t) \setminus s} m_{ut}(x_t)
\]

I) Message Product

II) Message Propagation
Belief Propagation for Trees

• Dynamic programming algorithm which exactly computes all marginals

• On Markov chains, BP equivalent to alpha-beta or forward-backward algorithms for HMMs

• Sequential *message schedules* require each message to be updated only once

• Computational cost:

  \[
  N \quad \text{number of nodes}
  \]

  \[
  M \quad \text{discrete states for each node}
  \]

  Belief Prop: \( \mathcal{O}(NM^2) \)

  Brute Force: \( \mathcal{O}(M^N) \)
Inference for Graphs with Cycles

• For graphs with cycles, the dynamic programming BP derivation breaks

**Junction Tree Algorithm**
• Cluster nodes to break cycles
• Run BP on the tree of clusters
• Exact, but often intractable

**Loopy Belief Propagation**
• Iterate local BP message updates on the graph with cycles
• Hope beliefs converge
• Empirically, often very effective…
A Brief History of Loopy BP

• 1993: Turbo codes (and later LDPC codes, rediscovered from Gallager’s 1963 thesis) revolutionize error correcting codes (Berrou et. al.)

• 1995-1997: Realization that turbo decoding algorithm is equivalent to loopy BP (MacKay & Neal)

• 1997-1999: Promising results in other domains, & theoretical analysis via computation trees (Weiss)

• 2000: Connection between loopy BP & variational approximations, using ideas from statistical physics (Yedidia, Freeman, & Weiss)

• 2001-2005: Many results interpreting, justifying, and extending loopy BP
Outline

**Inference in Graphical Models**
- Pairwise Markov random fields
- Belief propagation for trees

**Variational Methods**
- Mean field
- Bethe approximation & BP

**Extensions of Belief Propagation**
- Efficient message passing implementation
- Generalized BP
- Particle filters and nonparametric BP
Approximate Inference Framework

\[ p(x \mid y) = \frac{1}{Z} \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \prod_{s \in \mathcal{V}} \psi_s(x_s, y) \]

- Choose a family of approximating distributions which is tractable. The simplest example:
  \[ q(x) = \prod_{s \in \mathcal{V}} q_s(x_s) \]

- Define a distance to measure the quality of different approximations. Two possibilities:
  \[ D(p \parallel q) = \sum_x p(x \mid y) \log \frac{p(x \mid y)}{q(x)} \]
  \[ D(q \parallel p) = \sum_x q(x) \log \frac{q(x)}{p(x \mid y)} \]

- Find the approximation minimizing this distance
Fully Factored Approximations

\[ p(x \mid y) = \frac{1}{Z} \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \prod_{s \in \mathcal{V}} \psi_s(x_s, y) \]

\[ q(x) = \prod_{s \in \mathcal{V}} q_s(x_s) \]

\[ D(p \parallel q) = \sum_x p(x \mid y) \log \frac{p(x \mid y)}{q(x)} \]

\[ = \left[ \sum_{s \in \mathcal{V}} H_s(p_s) - H(p) \right] + \sum_{s \in \mathcal{V}} D(p_s \parallel q_s) \]

- Trivially minimized by setting \( q_s(x_s) = p_s(x_s \mid y) \)
- Doesn’t provide a computational method…
Variational Approximations

\[ D(q(x) \| p(x \mid y)) = \sum_x q(x) \log \frac{q(x)}{p(x \mid y)} \]

\[ \log p(y) = \log \sum_x p(x, y) \]

\[ = \log \sum_x q(x) \frac{p(x, y)}{q(x)} \]

\[ \geq \sum_x q(x) \log \frac{p(x, y)}{q(x)} \]

\[ = -D(q(x) \| p(x \mid y)) + \log p(y) \]

- Minimizing KL divergence maximizes a lower bound on the data likelihood
Free Energies

\[ p(x \mid y) = \frac{1}{Z} \exp \{-E(x)\} \]

\[ D(q \mid\mid p) = \sum_x q(x) \log q(x) - \sum_x q(x) \log p(x \mid y) \]

\[ = -H(q) + \sum_x q(x) E(x) + \log Z \]

- **Negative Entropy**
- **Average Energy**
- **Normalization**

**Gibbs Free Energy**

- Free energies equivalent to KL divergence, up to a normalization constant
Mean Field Free Energy

\[ p(x \mid y) = \frac{1}{Z} \exp \left\{ - \sum_{(s,t) \in \mathcal{E}} \phi_{st}(x_s, x_t) - \sum_{s \in \mathcal{V}} \phi_s(x_s, y) \right\} \]

\[ q(x) = \prod_{s \in \mathcal{V}} q_s(x_s) \]

\[ D(q \parallel p) = -H(q) + \sum_x q(x) E(x) + \log Z \]

\[ = -\sum_{s \in \mathcal{V}} H_s(q_s) + \sum_{(s,t) \in \mathcal{E}} q_s(x_s) q_t(x_t) \phi_{st}(x_s, x_t) \]

\[ \cdots + \sum_{s \in \mathcal{V}} q_s(x_s) \phi_s(x_s) + \log Z \]
Mean Field Equations

\[ D(q \parallel p) = - \sum_{s \in V} H_s(q_s) + \sum_{(s,t) \in E} q_s(x_s)q_t(x_t)\phi_{st}(x_s, x_t) \]
\[ \cdots + \sum_{s \in V} q_s(x_s)\phi_s(x_s) + \log Z \]

• Add Lagrange multipliers to enforce \[ \sum_{x_s} q_s(x_s) = 1 \]

• Taking derivatives and simplifying, we find a set of fixed point equations:

\[ q_s(x_s) = \alpha \psi_s(x_s) \prod_{t \in \Gamma(s)} \prod_{x_t} \psi_{st}(x_s, x_t)q_t(x_t) \]

• Updating one marginal at a time gives convergent coordinate descent
Structured Mean Field

- Any subgraph for which inference is tractable leads to a mean field style approximation for which the update equations are tractable.
Tree Structured Free Energies

- Trees exactly factorize as

\[ q(x) = \prod_{(s,t) \in \mathcal{E}} \frac{q_{st}(x_s, x_t)}{q_s(x_s)q_t(x_t)} \prod_{s \in \mathcal{V}} q_s(x_s) \]

- We may then optimize over all distributions which are Markov with respect to a tree-structured graph:

\[ D(q \mid p) = -H(q) + \sum_x q(x)E(x) + \log Z \]

\[ \sum_x q(x)E(x) = \sum_{(s,t) \in \mathcal{E}} q_{st}(x_s, x_t)\phi_{st}(x_s, x_t) + \sum_{s \in \mathcal{V}} q_s(x_s)\phi_s(x_s) \]

\[ H(q) = \sum_{s \in \mathcal{V}} H_s(q_s) - \sum_{(s,t) \in \mathcal{E}} I_{st}(q_{st}) \]

Marginal Entropies  
Mutual Information
Bethe Free Energy

- Bethe approximation uses the tree-structured free energy form even though the graph has cycles.

\[
D(q \parallel p) = -H(q) + \sum_x q(x)E(x) + \log Z
\]

**Average Energy (exact for pairwise MRFs)**

\[
\sum_x q(x)E(x) = \sum_{(s,t) \in \mathcal{E}} q_{st}(x_s, x_t)\phi_{st}(x_s, x_t) + \sum_{s \in \mathcal{V}} q_s(x_s)\phi_s(x_s)
\]

**Approximate Entropy**

\[
H(q) \approx \sum_{s \in \mathcal{V}} H_s(q_s) - \sum_{(s,t) \in \mathcal{E}} I_{st}(q_{st})
\]
Minimizing Bethe Free Energy

\[ D(q \parallel p) = -H(q) + \sum_x q(x)E(x) + \log Z \]

\[ \sum_x q(x)E(x) = \sum_{(s,t) \in E} q_{st}(x_s, x_t) \phi_{st}(x_s, x_t) + \sum_{s \in V} q_s(x_s) \phi_s(x_s) \]

\[ H(q) \approx \sum_{s \in V} H_s(q_s) - \sum_{(s,t) \in E} I_{st}(q_{st}) \]

- Add Lagrange multipliers to enforce normalizations:
  \[ \lambda_{st}(x_t) \leftrightarrow \sum_{x_s} q_{st}(x_s, x_t) = q_t(x_t) \quad \sum_{x_s} q_s(x_s) = 1 \]

- Taking derivatives and simplifying,
  \[ q_t(x_t) = \alpha \exp \left\{ \phi_t(x_t) + \frac{1}{|\Gamma(t)| - 1} \sum_{s \in \Gamma(t)} \lambda_{st}(x_t) \right\} \]
  \[ q_{st}(x_s, x_t) = \alpha \exp \{ \phi_{st}(x_s, x_t) + \phi_s(x_s) + \phi_t(x_t) + \lambda_{ls}(x_s) + \lambda_{st}(x_t) \} \]
Bethe and Belief Propagation

Bethe Fixed Points

\[ q_t(x_t) = \alpha \psi_t(x_t) \exp \left\{ \frac{1}{|\Gamma(t)| - 1} \sum_{s \in \Gamma(t)} \lambda_{st}(x_t) \right\} \]

\[ q_{st}(x_s, x_t) = \alpha \psi_{st}(x_s, x_t) \psi_s(x_s) \psi_t(x_t) \exp \{ \lambda_{ts}(x_s) + \lambda_{st}(x_t) \} \]

Belief Propagation

\[ q_t(x_t) = \alpha \psi_t(x_t, y) \prod_{u \in \Gamma(t)} m_{ut}(x_t) \]

\[ q_{st}(x_s, x_t) = \alpha \psi_{st}(x_s, x_t) \psi_s(x_s) \psi_t(x_t) \prod_{u \in \Gamma(s) \setminus t} m_{us}(x_s) \prod_{v \in \Gamma(t) \setminus s} m_{vt}(x_t) \]

\[ m_{ts}(x_s) = \alpha \sum_{x_t} \psi_{st}(x_s, x_t) \psi_t(x_t, y) \prod_{u \in \Gamma(t) \setminus s} m_{ut}(x_t) \]

Correspondence

\[ \lambda_{st}(x_t) = \log \prod_{u \in \Gamma(t) \setminus s} m_{ut}(x_t) \]
Implications for Loopy BP

Bethe Free Energy is an Approximation

- BP may have multiple fixed points (non-convex)
- BP is not guaranteed to converge
- Few general guarantees on BP’s accuracy

Characterizations of BP Fixed Points

- All graphical models have at least one BP fixed point
- Stable fixed points are local minima of Bethe
- For graphs with cycles, BP is almost never exact
- As cycles grow long, BP becomes exact (coding)
Outline

Inference in Graphical Models
- Pairwise Markov random fields
- Belief propagation for trees

Variational Methods
- Mean field
- Bethe approximation & BP

Extensions of Belief Propagation
- Efficient message passing implementation
- Generalized BP
- Particle filters and nonparametric BP
Double-Loop Algorithms
(Yuille & Rangarajan, Neural Comp. 2003)

\[ D(q \parallel p) = -H(q) + \sum_x q(x)E(x) + \log Z \]

\[ \sum_x q(x)E(x) = \sum_{(s,t) \in \mathcal{E}} q_{st}(x_s, x_t)\phi_{st}(x_s, x_t) + \sum_{s \in \mathcal{V}} q_s(x_s)\phi_s(x_s) \]

\[ H(q) \approx \sum_{s \in \mathcal{V}} H_s(q_s) - \sum_{(s,t) \in \mathcal{E}} I_{st}(q_{st}) \]

- Directly minimize Bethe free energy
- Guaranteed to converge to a local optimum
- Much slower than loopy BP
- Some theory and experimental results suggesting that when BP doesn’t converge, it’s a sign that Bethe approximation is bad
Efficient Message Updates

\[ m_{ts}(x_s) = \alpha \sum_{x_t} \psi_{st}(x_s, x_t) \psi_t(x_t, y) \prod_{u \in \Gamma(t) \setminus s} m_{ut}(x_t) \]

- Message update is matrix-vector product \( O(M^2) \)

- For pairwise potentials which depend only on the difference \( (x_s - x_t) \), this becomes a convolution
  - FFT message updates in \( O(M \log M) \)
  - Other approximations sometimes allow \( O(M) \)
Dynamic Quantization
(Coughlan et. al., ECCV 2002 & 2004 CVPR GMBV workshop)

Images removed due to copyright considerations.

• Deformable template: State at each node is discretization of position and orientation of some point along the letter contour

• Rules for pruning unlikely states based on local evidence, and current message estimates, allow efficient, nearly globally optimal matching
Generalized Belief Propagation
(Yedidia, Freeman, & Weiss, NIPS 2000)

- Big idea: cluster nodes to break shortest cycles
- Non-overlapping clusters: exactly equivalent to loopy BP on the graph of cluster nodes
- Overlapping clusters: higher-order Kikuchi free energies ensure that information not over-counted
BP for Continuous Variables

\[ q_t(x_t) = \alpha \psi_t(x_t, y) \prod_{u \in \Gamma(t)} m_{ut}(x_t) \]

\[ m_{ts}(x_s) = \alpha \int_{x_t} \psi_{st}(x_s, x_t) \psi_t(x_t, y) \prod_{u \in \Gamma(t) \setminus s} m_{ut}(x_t) \, dx_t \]

**Jointly Gaussian Variables**

- Messages represented by mean and covariance
- BP updates are generalizations of the Kalman filter
- If BP converges, means exact, but variances approximate

**Continuous Non-Gaussian Variables**

- Closed parametric forms usually do not exist
- Discretization can be intractable in as few as 2-3 dim.
Particle Filters

Condensation, Sequential Monte Carlo, Survival of the Fittest, …

- Nonparametric approximation to optimal BP estimates
- Represent messages and posteriors using a set of samples, found by simulation

Sample-based density estimate

Weight by observation likelihood

Resample & propagate by dynamics
Nonparametric Belief Propagation

(Sudderth, Ihler, Freeman, & Willsky)

**Belief Propagation**
- General graphs
- Discrete or Gaussian

**Particle Filters**
- Markov chains
- General potentials

**Nonparametric BP**
- General graphs
- General potentials