Logistics

Announcements

- Should have read chapters 1 through 5 in our book. Read chapter 7
- Also read chapter 8 (integer/linear programming, although we cover only a bit of that chapter in class unfortunately).
- Also should have read “Divergence measures and message passing” by Thomas Minka, and “Structured Region Graphs: Morphing EP into GBP”, by Welling, Minka, and Teh.
- Assignment due Wednesday (Dec 3rd) night, 11:45pm. Final project proposal final progress report (one page max).
- Update: For status update, final writeup, and talk, use notation as close as possible to that used in class!
On Final Project

- Project update report due tonight, 11:45pm via canvas.
- Final four-page writeup due **next Wednesday at 11:45pm**.
- Final writeup: 4-pages, 10 point font, 8.5x11 inch pages, 1 inch margins on all four sides.
- Again, all your writeups (starting tonight) should use notation as close as possible to what we’ve been using in class!
- Talk slides need to be uploaded before. Must be pdf, all will be merged into one pdf file. No animations.
- We have 21 presentations to give. 10 minutes each means 3.5 hours of presentation. 7 minutes each means 2.45 hours of presentation.
- Final Exam time slot: Wednesday, December 10, 2014, 2-4 pm, PCAR 297 (two hours).
- Alternatively, you each do a 10-minute youtube presentation with at least screen capture and audio, can use perhaps http://tinytake.com/ or http://camstudio.org/, or post your favorite to canvas for others to discover. Then, it to an unlisted youtube link, send the link, and we all view it.

Class Road Map - EE512a

- L1 (9/29): Introduction, Families, Semantics
- L2 (10/1): MRFs, elimination, Inference on Trees
- L3 (10/6): Tree inference, message passing, more general queries, non-tree
- L4 (10/8): Non-trees, perfect elimination, triangulated graphs
- L5 (10/13): triangulated graphs, k-trees, the triangulation process/heuristics
- L6 (10/15): multiple queries, decomposable models, junction trees
- L7 (10/20): junction trees, begin intersection graphs
- L8 (10/22): intersection graphs, inference on junction trees
- L9 (10/27): inference on junction trees, semirings
- L10 (11/3): conditioning, hardness, LBP
- L11 (11/5): LBP, exponential models,
- L12 (11/10): exponential models, mean params and polytopes,
- L13 (11/12): polytopes, tree outer bound, Bethe entropy approx.
- L14 (11/17): Bethe entropy approx, loop series correction
- L15 (11/19): Hypergraphs, posets, Mobius, Kikuchi
- L16 (11/24): Kikuchi, Expectation Propagation
- L17 (11/26): Expectation Propagation, Mean Field
- L18 (12/1): Structured mean field, Convex relaxations and upper bounds, tree reweighted case
- L19 (12/3): Variational MPE, Graph Cut MPE, LP Relaxations
- Final Presentations: (12/10):

Finals Week: Dec 8th-12th, 2014.
Conjugate Duality, Maximum Likelihood, Negative Entropy

Theorem 19.2.3 (Relationship between $A$ and $A^*$)

(a) For any $\mu \in \mathcal{M}^\circ$, $\theta(\mu)$ unique canonical parameter sat. matching condition, then conj. dual takes form:

$$A^*(\mu) = \sup_{\theta \in \Omega} (\langle \theta, \mu \rangle - A(\theta)) = \begin{cases} -H(p_{\theta(\mu)}) & \text{if } \mu \in \mathcal{M}^\circ \\ +\infty & \text{if } \mu \notin \mathcal{M} \end{cases} \quad (19.3)$$

(b) Partition function has variational representation (dual of dual)

$$A(\theta) = \sup_{\mu \in \mathcal{M}} \{\langle \theta, \mu \rangle - A^*(\mu)\} \quad (19.4)$$

(c) For $\theta \in \Omega$, sup occurs at $\mu \in \mathcal{M}^\circ$ of moment matching conditions

$$\mu = \int_{D_X} \phi(x)p_{\theta}(x)\nu(dx) = \mathbb{E}_{\theta}[\phi(X)] = \nabla A(\theta) \quad (19.5)$$

Variational Approach Amenable to Approximation

Variational Approximations we cover

- Original variational representation of log partition function

$$A(\theta) = \sup_{\mu \in \mathcal{M}} \{\langle \theta, \mu \rangle - A^*(\mu)\} \quad (19.1)$$

where dual takes form:

$$A^*(\mu) = \sup_{\theta \in \Omega} (\langle \theta, \mu \rangle - A(\theta)) = \begin{cases} -H(p_{\theta(\mu)}) & \text{if } \mu \in \mathcal{M}^\circ \\ +\infty & \text{if } \mu \notin \mathcal{M} \end{cases} \quad (19.2)$$

- Given efficient expression for $A(\theta)$, we can compute marginals of interest.
- Above expression (dual of the dual) offers strategies to approximate or (upper or lower) bound $A(\theta)$. We either approximate $\mathcal{M}$ or $-A^*(\mu)$ or (most likely) both.

1. Set $\mathcal{M} \leftarrow \mathbb{L}$ and $-A^*(\mu) \leftarrow H_{\text{Bethe}}(\tau)$ to get Bethe variational approximation, LBP fixed point.
2. Set $\mathcal{M} \leftarrow \mathbb{L}(G)$ (hypergraph marginal polytope), $-A^*(\mu) \leftarrow H_{\text{app}}(\tau)$ where $H_{\text{app}} = \sum_{q \in E} L_q g(q)$ (via M"obius) to get Kikuchi variational approximation, message passing on hypergraphs.
3. Partition function $(-\infty)$ used to find $\mathbb{L}(\lambda)$ value.
Variational Approach Amenable to Approximation

Variational Approximations we cover

- Original variational representation of log partition function
  \[ A(\theta) = \sup_{\mu \in \mathcal{M}} \{ \langle \theta, \mu \rangle - A^*(\mu) \} \tag{19.1} \]
  where dual takes form:
  \[ A^*(\mu) = \sup_{\theta \in \Omega} (\langle \theta, \mu \rangle - A(\theta)) = \begin{cases} -H(p_{\theta}(\mu)) & \text{if } \mu \in \mathcal{M}^c \\ +\infty & \text{if } \mu \notin \mathcal{M} \end{cases} \tag{19.2} \]
- Given efficient expression for \( A(\theta) \), we can compute marginals of interest.
- Above expression (dual of the dual) offers strategies to approximate or (upper or lower) bound \( A(\theta) \). We either approximate \( \mathcal{M} \) or \( -A^*(\mu) \) or (most likely) both.
  1. Set \( \mathcal{M} \leftarrow \mathbb{L} \) and \( -A^*(\mu) \leftarrow H_{\text{Bethe}}(\tau) \) to get Bethe variational approximation, LBP fixed point.
  2. Set \( \mathcal{M} \leftarrow \mathbb{L}_t(G) \) (hypergraph marginal polytope), \( -A^*(\mu) \leftarrow H_{\text{app}}(\tau) \) where \( H_{\text{app}} = \sum_{g \in \mathcal{E}} c(g) H_g(\tau_g) \) (via M"obius) to get Kikuchi variational approximation, message passing on hypergraphs.
  3. Partition \( \tau \) into \( (\tau, \tilde{\tau}) \), and set \( \mathcal{M} \leftarrow \mathcal{L}(\phi, \Phi) \) and set \( -A^*(\mu) \leftarrow H_{\text{ep}}(\tau, \tilde{\tau}) \) to get expectation propagation.

MPE - most probable explanation

- In many cases, we care not to sum over \( x \) in \( \sum_x p(x) \) but instead to compute \( x^* \in \arg\max_{x \in D_X} p(x) \).
- This is called the “Viterbi assignment”, or the “most probable explanation” (MPE), or the “most probable configuration” or the “mode”, or a few other names.
- From the perspective of semirings, we are only changing the semiring (from sum-product to max-product). Can do exactly same form of exact inference algorithms (e.g., trees, \( k \)-trees, junction trees) using different semiring, to get answer. To get \( n \)-best answers, can also be seen as a semiring.
- Equally difficult when tree-width is large.
- Can the variational approach help in this case as well?
MPE - most probable explanation

- MPE again

\[
\arg\max_{x \in X^m} p(x) = \{ x \in D^m : p_\theta(x) \geq p_\theta(y), \forall y \in D^m \} \quad (19.1)
\]

- Since we are using exponential family models, we have

\[
\arg\max_{x \in X^m} p(x) = \arg\max_{x \in X^m} \langle \theta, \phi(x) \rangle = \arg\min_{x \in X^m} E[x] \quad (19.2)
\]

i.e., cumulant function isn’t required for computation.

\( E[x] = -\langle \theta, \phi(x) \rangle \) is seen as an “energy” function.

- But it is related. Recall cumulant function

\[
A(\theta) = \log \int \exp \{ \langle \theta, \phi(x) \rangle \} d\nu(x) \quad (19.3)
\]

\[
= \sup_{\mu \in M} \{ \langle \theta, \mu \rangle - A^*(\mu) \} \quad (19.4)
\]

MPE - and variational

- Considering \( p_\theta(x) = \exp \{ \langle \theta, \phi(x) \rangle - A(\theta) \} \).

- Let \( \beta \in \mathbb{R}_+ \) be a positive scalar.

- If we substitute \( \theta \) with \( \beta \theta \) (i.e., \( p_\theta(x) \) with \( p_{\beta \theta}(x) \)), and when \( \beta \theta \in \Omega \),
then \( p_{\beta \theta}(x) \) becomes more concentrated (relatively) around MPE solutions as \( \beta \to \infty \).

- Ex: Let \( p_\theta(x^*) > p_\theta(y) \) for all \( y \neq x^* \), so \( x^* \) is the unique maximum. Then \( \langle \theta, \phi(x^*) \rangle > \langle \theta, \phi(y) \rangle \) and

\[
h(\beta) \triangleq \langle \beta \theta, \phi(x^*) \rangle - \langle \beta \theta, \phi(y) \rangle = \beta(\langle \theta, \phi(x^*) \rangle - \langle \theta, \phi(y) \rangle) \quad (19.5)
\]

grows unboundedly large as \( \beta \to \infty \).

- Since \( A(\beta \theta) \) keeps things normalized, \( A(\beta \theta) \) somehow must counteract the otherwise unbounded increase in \( h(\beta) \). This suggests \( A(\beta \theta)/\beta \) might tell us something.
**Theorem 19.3.1 (MPE and variational)**

For all $\theta \in \Omega$, the problem of mode computation has the following alternative representations:

$$
\max_{x \in D_{X^m}} \langle \theta, \phi(x) \rangle = \max_{\mu \in \bar{M}} \langle \theta, \mu \rangle, \quad \text{and} \quad (19.6)
$$

$$
\max_{x \in D_{X^m}} \langle \theta, \phi(x) \rangle = \lim_{\beta \to \infty} \frac{A(\beta \theta)}{\beta} \quad (19.7)
$$

- **Intuition:** We have $\mu = E_p[\phi(x)]$, so that $\max_{x \in D_{X^m}} \langle \theta, \phi(x) \rangle = \max_{p \in \mathcal{P}} \langle \theta, E_p[\phi(x)] \rangle$ where $\mathcal{P}$ is a set of zero entropy distributions with point mass on some point in $D_{X^m}$. I.e., for each $p \in \mathcal{P}$, there exists $x \in D_{X^m}$ with $p(x) = 1$.

- Equation (19.6) says that max falls on extreme point of the mean parameter convex region $\bar{M}$ (vertex of polytope, in polyhedral case).

---

**MPE - and variational**

- Also, Equation (19.6) shows how MPE can be seen as a linear optimization over a convex set $\mathcal{M}$.
- For discrete distributions, we have $\mathcal{M} = \mathbb{M}(G)$ for graph $G$, so this is a linear objective with polyhedral constraints, i.e., a linear program (LP).
- Since l.h.s. of Equation (19.6) is integer program, this shows the difficulty of $\mathbb{M}(G)$. 

MPE - and variational

- Intuition for Equation (19.7), repeated here:

$$\max_{x \in \mathcal{D}_X} \langle \theta, \phi(x) \rangle = \lim_{\beta \to \infty} \frac{A(\beta \theta)}{\beta}$$  \hfill (19.7)

- Intuitively,

$$\lim_{\beta \to +\infty} \frac{A(\beta \theta)}{\beta} = \lim_{\beta \to +\infty} \frac{1}{\beta} \sup_{\mu \in \mathcal{M}} \{\langle \beta \theta, \mu \rangle - A^*(\mu)\}$$  \hfill (19.8)

$$= \lim_{\beta \to +\infty} \sup_{\mu \in \mathcal{M}} \{\langle \theta, \mu \rangle - \frac{1}{\beta} A^*(\mu)\}$$  \hfill (19.9)

- Due to convexity of $A^*$ we can swap the $\lim$ and the $\sup$ and we get the result.

MPE - and variational for trees

- When graph is a tree, we can find an interesting connection between the max-product form of messages and a particular Lagrangian.

- Maxproduct updates take the form:

$$M_{t \rightarrow s}(x_s) \leftarrow \kappa \max_{x_t \in \mathcal{D}_X} \left[ \exp \left\{ \theta_{st}(x_s, x_t) + \theta_t(x_t) \right\} \prod_{u \in N(t) \setminus s} M_{u \rightarrow t}(x_t) \right]$$  \hfill (19.10)

- Using the Theorem 19.3.1, we get (in the case of a tree $T$)

$$\max_{x \in \mathcal{D}_X^m} \left[ \sum_{s \in \mathcal{V}} \theta_s(x_s) + \sum_{(s,t) \in \mathcal{E}} \theta_{st}(x_s, x_t) \right] = \max_{\mu \in \mathcal{L}(T)} \langle \mu, \theta \rangle$$  \hfill (19.11)

- Right hand side is a LP over a simple polytope, the marginal polytope for trees $\mathcal{L}(T)$. 
MPE, relationship between max-product algorithm and linear program

- It turns out that: the max-product updates are a Lagrangian method for solving the dual of the above linear program, i.e., \( \max_{\mu \in \mathbb{L}(T)} \langle \mu, \theta \rangle \).
- Marginalization constraint \( C_{ts}(x_s) = 0 \) for edge \( t, s \)

\[
C_{ts}(x_s) = \mu_s(x_s) - \sum_{x_t} \mu_{st}(x_s, x_t)
\] (19.12)

and associated Lagrange multiplier \( \lambda_{st}(x_s) \).
- Also define a (non-negative and normalized) mean parameter space \( \mathbb{N} \subseteq \mathbb{R}^d \) as follows:

\[
\mathbb{N} = \left\{ \mu \in \mathbb{R}^d | \mu \geq 0, \sum_{x_s} \mu_s(x_s) = 1, \sum_{x_s, x_t} \mu_{st}(x_s, x_t) = 1 \right\}
\] (19.13)

Max-Product and LP Duality

**Theorem 19.3.2 (Max-product and LP Duality)**

Consider the dual function \( Q \) defined by the following partial Lagrangian formulation of the tree-structured LP:

\[
Q(\lambda) = \max_{\mu \in \mathbb{N}} L(\mu; \lambda), \quad \text{where}
\] (19.14)

\[
L(\mu; \lambda) = \langle \theta, \mu \rangle + \sum_{(s,t) \in E(T)} \left[ \sum_{x_s} \lambda_{ts}(x_s) C_{ts}(x_s) + \sum_{x_t} \lambda_{st}(x_t) C_{st}(x_t) \right]
\] (19.15)

For any fixed point \( M^* \) of the max-product updates, the vector \( \lambda^* = \log M^* \), where the logarithm is taken elementwise, is an optimal solution of the dual problem \( \min_{\lambda} Q(\lambda) \).
Here we don’t restrict $G$ but restrict clique functions.

Given $G$ let $p \in \mathcal{F}(G, \mathcal{M}(f))$ such that we can write

$$\log p(x) = \prod_{v \in V(G)} \psi_v(x_v) \prod_{(i,j) \in E(G)} \psi_{ij}(x_i, x_j)$$

(19.16)

or equivalently

$$-\log p(x) \propto \sum_{v \in V(G)} e_v(x_v) + \sum_{(i,j) \in E(G)} e_{ij}(x_i, x_j)$$

(19.17)

$e_v(x_v)$ and $e_{ij}(x_i, x_j)$ are like local energy potentials, the smaller they are, the higher the probability. E.g., $e_{ij}(x_i, x_j) = -\theta_{ij} \phi_{ij}(x_i, x_j)$

Restricted clique functions

Given $G$ let $p \in \mathcal{F}(G, \mathcal{M}(f))$ such that we can write the global energy $E(x)$ as a sum of unary and pairwise potentials:

$$E(x) = \sum_{v \in V(G)} e_v(x_v) + \sum_{(i,j) \in E(G)} e_{ij}(x_i, x_j)$$

(19.18)

e_v(x_v)$ and $e_{ij}(x_i, x_j)$ are like local energy potentials.

Since $\log p(x) = -E(x) + \text{const.}$, the smaller $e_v(x_v)$ or $e_{ij}(x_i, x_j)$ become, the higher the probability becomes.

Further, say that $D_{X_v} = \{0, 1\}$ (binary), so we have binary random vectors distributed according to $p(x)$.

Thus, $x \in \{0, 1\}^V$, and finding MPE solution is setting some of the variables to 0 and some to 1, i.e.,

$$\min_{x \in \{0, 1\}^V} E(x)$$

(19.19)
MRF example

Markov random field

\[
\log p(x) \propto \sum_{v \in V(G)} e_v(x_v) + \sum_{(i,j) \in E(G)} e_{ij}(x_i, x_j)
\]  (19.20)

When \( G \) is a 2D grid graph, we have

Create an auxiliary graph

- We can create auxiliary graph \( G_a \) that involves two new “terminal” nodes \( s \) and \( t \) and all of the original “non-terminal” nodes \( v \in V(G) \).
- The non-terminal nodes represent the original random variables \( x_v, v \in V \).
- Starting with the original grid-graph amongst the vertices \( v \in V \), we connect each of \( s \) and \( t \) to all of the original nodes.
- I.e., we form \( G_a = (V \cup \{s,t\}, E \cup \bigcup_{v \in V}(\{s,v\} \cup \{v,t\})) \).
Transformation from graphical model to auxiliary graph

Original 2D-grid graphical model $G$ and energy function $E(x) = \sum_{v \in V(G)} e_v(x_v) + \sum_{(i,j) \in E(G)} e_{ij}(x_i, x_j)$, needing to be minimized over $x \in \{0, 1\}^V$. Recall, tree-width is $O(\sqrt{|V|})$.

Augmented (graph-cut) directed graph $G_a$. Edge weights (TBD) of graph are derived from $\{e_v(\cdot)\}_{v \in V}$ and $\{e_{ij}(\cdot, \cdot)\}_{(i,j) \in E(G)}$.

A $(s, t)$-cut $C \subseteq E(G_a)$ is a set of edges that cut all paths from $s$ to $t$.

A minimum $(s, t)$-cut is one that has minimum weight, where $w(C) = \sum_{e \in C} w_e$ is the cut weight.

To be a cut, must have that, for every $v \in V$,

- either $(s, v) \in C$ or
- $(v, t) \in C$, directed, arrows pointing down from $s$ towards $t$ or from $i \rightarrow j$.

Cut edges that are incident to terminal nodes $s$ and $t$ are indicated in green. Cut edges that are incident to terminal nodes $s$ and $t$ removed from graph. But there are still un-cut $(s, t)$-paths remaining. Additional cut edges incident to two non-terminal nodes are indicated in green. Vertices adjacent to $t$ are shaded blue, vertices adjacent to $s$ shaded red. Additional cut edges incident to two non-terminal nodes are removed from graph. Augmented graph-cut graph with cut edges removed corresponds to particular binary vector $\bar{x} \in \{0, 1\}^n$.

Each vector $\bar{x}$ has a score corresponding to $\log p(\bar{x})$, but when can graph cut scores correspond precisely to $\log p(\bar{x})$ in a way that min-cut algorithms can find minimum of $E(x)$?

Setting of the weights in the auxiliary cut graph

- Any graph cut corresponds to a vector $\bar{x} \in \{0, 1\}^n$.
- If weights of all edges, except those involving terminals $s$ and $t$, are non-negative, graph cut computable in polynomial time via max-flow (many algorithms, e.g., Edmonds&Karp $O(nm^2)$ or $O(n^2m \log(nC))$; Goldberg&Tarjan $O(nm \log(n^2/m))$, see Schrijver, page 161).
- If weights are set correctly in the cut graph, and if edge functions $e_{ij}$ satisfy certain properties, then graph-cut score corresponding to $\bar{x}$ can be made equivalent to $E(x) = \log p(\bar{x}) + \text{const.}$.
- Hence, poly time graph cut, can find the optimal MPE assignment, regardless of the graphical model’s tree-width!
- In general, finding MPE is an NP-hard optimization problem.
Setting of the weights in the auxiliary cut graph

Edge weight assignments. Start with all weights set to zero.

- For $(s, v)$ with $v \in V(G)$, set edge
  \[ w_{s,v} = (e_v(1) - e_v(0))1(e_v(1) > e_v(0)) \]  
  (19.21)

- For $(v, t)$ with $v \in V(G)$, set edge
  \[ w_{v,t} = (e_v(0) - e_v(1))1(e_v(0) \geq e_v(1)) \]  
  (19.22)

- For original edge $(i,j) \in E$, $i, j \in V$, set weight
  \[ w_{i,j} = e_{ij}(1,0) + e_{ij}(0,1) - e_{ij}(1,1) - e_{ij}(0,0) \]  
  (19.23)

  and if $e_{ij}(1,0) > e_{ij}(0,0)$, and $e_{ij}(1,1) > e_{ij}(0,1)$,
  \[ w_{s,i} \leftarrow w_{s,i} + (e_{ij}(1,0) - e_{ij}(0,0)) \]  
  (19.24)

  \[ w_{j,t} \leftarrow w_{j,t} + (e_{ij}(1,1) - e_{ij}(0,1)) \]  
  (19.25)

  and analogous increments if inequalities are flipped.

Non-negative edge weights

- The inequalities ensure that we are adding non-negative weights to each of the edges. I.e., we do $w_{s,i} \leftarrow w_{s,i} + (e_{ij}(1,0) - e_{ij}(0,0))$ only if $e_{ij}(1,0) > e_{ij}(0,0)$.

- For $(i,j)$ edge weight, it takes the form:
  \[ w_{i,j} = e_{ij}(1,0) + e_{ij}(0,1) - e_{ij}(1,1) - e_{ij}(0,0) \]  
  (19.26)

- For this to be non-negative, we need:
  \[ e_{ij}(1,0) + e_{ij}(0,1) \geq e_{ij}(1,1) - e_{ij}(0,0) \]  
  (19.27)

- Thus weights $w_{ij}$ in $s,t$-graph above are always non-negative, so graph-cut solvable exactly.
Submodular potentials

- Edge functions must be submodular (in the binary case, equivalent to “associative”, “attractive”, “regular”, “Potts”, or “ferromagnetic”): for all \((i, j) \in E(G)\), must have:
  \[ e_{ij}(0,1) + e_{ij}(1,0) \geq e_{ij}(1,1) + e_{ij}(0,0) \]  
  (19.28)

- This means: on average, preservation is preferred over change.

- Actual probability are of the form \(p(x) \propto \prod \psi\), so this means \(\psi_{ij}(1,0)\psi_{ij}(0,1) \leq \psi_{ij}(0,0)\psi_{ij}(1,1)\): geometric mean of factor scores higher when neighboring pixels have the same value - a reasonable assumption about natural scenes and signals.

- As a set function, this is the same as:
  \[ f(X) = \sum_{\{i,j\} \in E(G)} f_{i,j}(X \cap \{i,j\}) \]  
  (19.29)

  which is submodular if each of the \(f_{i,j}\)'s are submodular!

- A special case of more general submodular functions – unconstrained submodular function minimization is solvable in polytime.

Theorem 19.4.1

If the edge functions are submodular and the edge weights in the \(s,t\)-graph are set as above, then finding the minimum \(s,t\)-cut in the auxiliary graph will yield a variable assignment having maximum probability.

Theorem 19.4.2

Submodular pairwise potentials is a necessary and sufficient condition for an energy function like the above \(E(x)\) to be graph representable, meaning that we can set up a graph cut based MPE inference algorithm and the resulting graph cut solves the MPE problem, \(\min_{x \in \{0,1\}^V} E(x) = \max_{x \in \{0,1\}^V} p(x)\), exactly in polytime in \(n = |V|\).

Proof.

See Kolmogorov 2004
Useful for computer vision

- Image segmentation problems can use such a model.
- Consider a 2D image, with a MRF to encode “smoothness” (i.e., spatial locality means things are likely to be similar).
- On average, similar neighbors have lower energy (higher probability) via
  \[ e_{ij}(0, 1) + e_{ij}(1, 0) \geq e_{ij}(1, 1) + e_{ij}(0, 0) \]

Graph Cut Marginalization

- What to do when potentials are not submodular? QPBO, quadratic pseudo Boolean optimization (computes only a partial solution).
- For non-binary, use move making algorithms (\(\alpha - \beta\)-swaps, \(\alpha\)-expansions, fusion moves, etc.)
- Is submodularity sufficient to make standard marginalization possible?
- Unfortunately, even in submodular case, computing partition function is a \(\#P\)-complete problem (if it was possible to do it in poly time, that would require \(P = NP\)).
- On the other hand, for pairwise MRFs, computing partition function in submodular potential case is approximable (has low error with high probability).
- Attractive potentials (generalization of submodular to non-binary case) leads to bound in Bethe, as we saw.
Bounds on inner product

- We know $\mathbb{L}(G) \supseteq \mathbb{M}(G)$ with equality only when $G = T$.
- Thus,

$$\max_{x \in \mathbb{D} \times m} \langle \theta, \phi(x) \rangle = \max_{\mu \in \mathbb{M}(G)} \langle \theta, \mu \rangle \leq \max_{\tau \in \mathbb{L}(G)} \langle \theta, \tau \rangle \quad (19.30)$$

- r.h.s. is called a first-order LP relaxation (i.e., due to 1-tree), with only linear number of constraints and can be solved exactly.
- Note, middle case means that solution lies on integral extremal point of polytope $\mathbb{M}(G)$ (always at least one extremal point in solution set of any LP over a polytope).
- i.e., solution is some point $\phi(y) = \mu_y \in \mathbb{M}(G)$ for solution vector $y \in \{0, 1\}^n$.
- We can relate extreme points of $\mathbb{M}(G)$ and $\mathbb{L}(G)$.

Extreme points

Proposition 19.5.1

The extreme points of $\mathbb{L}(G)$ and $\mathbb{M}(G)$ are related in the following way:

(a) All extreme points of $\mathbb{M}(G)$ are integral, each one is also an extreme point of $\mathbb{L}(G)$.

(b) For graphs with cycles, $\mathbb{L}(G)$ also includes additional extreme points with fractional elements that lie strictly outside of $\mathbb{M}(G)$.

- If the relaxation works or not, depends on the tightness. If we end up with integral point, we are tight and have an exact solution.
- If we end up with a fractional solution, we are not tight and instead are outside of $\mathbb{M}(G)$ and thus have only an approximate solution.
- In such case, we could potentially round the nonintegral values back down to integers.
Fractional solutions

- Perhaps fractional solutions have at least some information about the optimal solution.
- We get:

**Definition 19.5.2**

Given a fractional solution $\tau$ to the LP relaxation, let $I \subset V$ represent the subset of vertices for which $\tau_s$ has only integral elements, say fixing $x_s = x^*_s$ for all $s \in I$. The fractional solution is said to be **strongly persistent** if any optimal integral solution $y^*$ satisfies $y^*_s = x^*_s$ for all $s \in I$. The fractional solution is **weakly persistent** if there exists at least one optimal $y^*$ such that $y^*_s = x^*_s$ for all $s \in I$.

- So if either of these are true, we’d get some sort of partial solution.
- Strongly persistent ensures that no solutions are eliminated by sticking with the integral values of $x_s$ for $s \in I$.

**Persistent solutions in LP relaxation binary case**

**Proposition 19.5.3**

Suppose that the first-order LP relaxation is applied to the binary quadratic program

$$\max_{x \in \{0,1\}^m} \left\{ \sum_{s \in V} \theta_s x_s + \sum_{(s,t) \in E} \theta_{st} x_s x_t \right\} \quad (19.31)$$

Then any fractional solution is strongly persistent!
Higher order relaxations

- As you can imagine, higher order relaxations are possible.
- Kikuchi style relaxations, where pseudo marginals come from being consistent w.r.t. a graph other than a tree.
- Analogous to previous cases, could use a $k$-tree for $k > 1$ or define polytope based on being locally consistent w.r.t. some clustered instance, i.e., hypergraph.
- In each case, we'll get an upper bound approximation of the MPE problem.
- In each case, we'll have a Lagrangian, and can define max-marginal style messages that, if they converge, correspond to a fixed point.
- Important to generalize to discrete non-binary case, so far little is known (much work here done in the graph cut case, in terms of move-making algorithms).
- Can move-making algorithms be seen in the variational framework (i.e., is there a variational approximation such that move making algorithms correspond to fixed point of some Lagrangian?).

Graphical Model Inference

- We started by marginalizing variables, the elimination algorithm.
- Elimination couples variables together if the graph is not a tree.
- All graphs can be embedded into a hypertree if the “width” of the tree is wide enough.
- Want to find slimmest possible tree into which a graph can be embedded.
- Once done we can convert to junction tree and run message passing (equivalent to eliminating on the hypertree).
- Often, slimmest possible tree (even if we could find it) is not slim enough, need approximation.
Approximation: Two general approaches

- exact solution to approximate problem - approximate problem
  - learning with or using a model with a structural restriction, structure learning, using a $k$-tree for a lower $k$ than one knows is true. Make sure $k$ is small enough so that exact inference can be performed, and make sure that, in that low tree-width model, one has best possible graph
  - Functional restrictions to the model (i.e., use factors or potential functions that obey certain properties). Then certain fast algorithms (e.g., graph-cut) can be performed.

- approximate solution to exact problem - approximate inference
  - Message or other form of propagation, variational approaches, LP relaxations, loopy belief propagation (LBP)
  - sampling (Monte Carlo, MCMC, importance sampling) and pruning (e.g., search based A*, score based, number of hypothesis based) procedures
  - Both methods only guaranteed approximate quality solutions.
  - No longer in the achievable region in time-space tradeoff graph, new set of time/space tradeoffs to achieve a particular accuracy.
Theorem 19.6.3 (Relationship between $A$ and $A^*$)

(a) For any $\mu \in \mathcal{M}^\circ$, $\theta(\mu)$ unique canonical parameter s.t. matching condition, then conj. dual takes form:

$$A^*(\mu) = \sup_{\theta \in \Omega} \langle \theta, \mu \rangle - A(\theta) = \begin{cases} -H(p_{\theta(\mu)}) & \text{if } \mu \in \mathcal{M}^\circ \\ +\infty & \text{if } \mu \notin \overline{\mathcal{M}} \end{cases}$$  \hspace{1cm} (19.3)

(b) Partition function has variational representation (dual of dual)

$$A(\theta) = \sup_{\mu \in \mathcal{M}} \{ \langle \theta, \mu \rangle - A^*(\mu) \}$$  \hspace{1cm} (19.4)

(c) For $\theta \in \Omega$, sup occurs at $\mu \in \mathcal{M}^\circ$ of moment matching conditions

$$\mu = \int_{D_X} \phi(x)p_\theta(x)\nu(dx) = E_\theta[\phi(X)] = \nabla A(\theta)$$  \hspace{1cm} (19.5)
Variational Approach Amenable to Approximation

Variational Approximations we cover

- Original variational representation of log partition function
  \[ A(\theta) = \sup_{\mu \in \mathcal{M}} \{ \langle \theta, \mu \rangle - A^*(\mu) \} \]  
  where dual takes form:
  \[ A^*(\mu) = \sup_{\theta \in \Omega} (\langle \theta, \mu \rangle - A(\theta)) = \begin{cases} 
  -H(p_{\theta}(\mu)) & \text{if } \mu \in \mathcal{M}^\circ \\
  +\infty & \text{if } \mu \notin \mathcal{M}
  \end{cases} \]  

- Given efficient expression for \( A(\theta) \), we can compute marginals of interest.

- Above expression (dual of the dual) offers strategies to approximate or (upper or lower) bound \( A(\theta) \). We either approximate \( \mathcal{M} \) or \(-A^*(\mu)\) or (most likely) both.

1. Set \( \mathcal{M} \leftarrow \mathbb{L} \) and \(-A^*(\mu) \leftarrow H_{\text{Bethe}}(\tau) \) to get Bethe variational approximation, LBP fixed point.

2. Set \( \mathcal{M} \leftarrow \mathbb{L}_t(G) \) (hypergraph marginal polytope), \(-A^*(\mu) \leftarrow H_{\text{app}}(\tau) \) where \( H_{\text{app}} = \sum_{g \in \mathcal{B}(G)} H_g(\tau_g) \) (via Mobius) to get Kikuchi variational approximation, message passing on hypergraphs.

3. Partition \( \tau \) into \((\tau, \tilde{\tau})\), and set \( \mathcal{M} \leftarrow \mathcal{L}(\phi, \Phi) \) and set \(-A^*(\mu) \leftarrow H_{\text{ep}}(\tau, \tilde{\tau}) \) to get expectation propagation.

Sources for Today’s Lecture


- Earlier lectures of this class.