EE12A – Advanced Inference in Graphical Models
Fall 2011

Prof. Jeff Bilmes

University of Washington, Seattle
Department of Electrical Engineering
Fall Quarter, 2011
http://j.ee.washington.edu/~bilmes/classes/ee512a_fall_2011/

Lecture 10 - Nov 2nd, 2011

Outstanding Reading

- Read chapters 1, 2, and 3 in the Wainwright and Jordan book.
We need to find one makeup lecture this term.

- **L1 (9/28):** Introduction, Families, Semantics
- **LX (9/30):** No class
- **L2 (10/5):** Trees, exact inference
- **L3 (10/7):** More on trees and inference.
- **L4 (10/12):** To tree or not to tree.
- **L5 (10/14):** All models lead to trees
- **L6 (10/19):** Decomposable, JT
- **L7 (10/21):** Inference on JTs
- **L8 (10/26):** JT Inference, semi-rings,
- **L9 (10/28):** time-space tradeoff, conditioning, LBP
- **L10 (11/2):** LBP, exp. f. models
- **L11 (11/4):**
- **L12 (11/9):**
- **LXX (11/11):** Veterans Day, no class
- **L13 (11/16):**
- **L14 (11/18):**
- **L15 (11/23):**
- **LXX (11/25):** Thanksgiving, no class
- **L16 (11/30):**
- **L17 (12/2):**
- **L18 (12/7):**
- **L19: (12/9):**

**Review**

Time and memory complexity is $O(Nr^{\omega+1})$ where $\omega$ is the tree-width.

We can use conditioning (e.g., cutset conditioning) to get other points. E.g., condition on a set that renders the remainder of the set a tree. Same computation less memory.

Recursive conditioning allows is to get linear memory but a time complexity of $O(r^{(w+1)\log N})$.

In general, many time-space tradeoffs for exact inference. Many algorithms along the achievable/unachievable frontier are SAT/CSP based, and use conditioning combined with various caching, and clause learning/deduction (e.g., nogood learning).

To get a better time/space profile, need to do approximation.

For any given degree of distortion, there is a time/space tradeoff profile.
### 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
  - Sampling
  - etc.

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.

### Belief Propagation

**Generic message definition**

\[
\mu_{i \rightarrow j}(x_j) = \sum_{x_i} \psi_{i,j}(x_i, x_j) \prod_{k \in \delta(i) \setminus \{j\}} \mu_{k \rightarrow i}(x_i)
\]  

(1)

- If graph is a tree, and if we obey MPP order, then we will reach a point where we’ve got marginals. I.e.,

\[
p(x_i) \propto \prod_{j \in \delta(i)} \mu_{j \rightarrow i}(x_i)
\]

(2)

and

\[
p(x_i, x_j) \propto \psi_{i,j}(x_i, x_j) \prod_{k \in \delta(i) \setminus \{j\}} \mu_{k \rightarrow i}(x_i) \prod_{\ell \in \delta(j) \setminus \{i\}} \mu_{\ell \rightarrow j}(x_j)
\]

(3)
Belief Propagation

Often, we see that nodes have potential functions as well. I.e., we have edge potentials \( \psi_{i,j}(x_i, x_j) \) for \((i,j) \in E(G)\) and \( \psi_i(x_i) \) for \( i \in V(G) \). Also, we might normalize each step. We get:

\[
\mu_{i \rightarrow j}(x_j) \propto \sum_{x_i} \psi_{i,j}(x_i, x_j) \psi_i(x_i) \prod_{k \in \delta(i) \setminus \{j\}} \mu_{k \rightarrow i}(x_i)
\]

such that \( \mu_{i \rightarrow j}(x_j) \) sums to 1. If \( G \) is a tree, and we obey MPP, we get

\[
p(x_i) \propto \psi_i(x_i) \prod_{j \in \delta(i)} \mu_{j \rightarrow i}(x_i)
\]

and

\[
p(x_i, x_j) \propto \psi_{i,j}(x_i, x_j) \prod_{k \in \delta(i) \setminus \{j\}} \mu_{k \rightarrow i}(x_i) \prod_{\ell \in \delta(j) \setminus \{i\}} \mu_{\ell \rightarrow j}(x_j)
\]

Reparameterization

- We start with a general \( p \in \mathcal{F}(G, R^f) \) in terms of factors that might not alone have any inherent meaning or normalization.
- Goal: We reparametrize \( p \) so that the factor decomposition is the same but the factors are now marginals – marginal reparameterization.
- Tree graph is such that we can reparameterize so that the edges and nodes are true marginals. E.g., \( \phi_i(x_i) = \sum_{x_V \setminus \{i\}} p(x) \).
- Can we always do this? Only when graph is triangulated and we do it in terms of cliques and separators. When graph is not triangulated, not possible in general to do this. E.g., 4-cycle.
Reparameterization

- In a tree, we achieve true marginal reparameterization by sending messages according to MPP until all messages are sent in both directions.
- Alternatively, we could, say, initialize all messages to unity $\mu_{i \rightarrow j}(x_j) = 1$ or some other set of values, and sending all messages in parallel. Each parallel send of all message is considered one step.
- Let $D$ be the diameter of the tree (length of longest path).
- Once we have done $D$ steps, we will have “converged.” Any additional messages will not change the state.
- If we have a tree, we have achieved marginal reparameterization.

State representation

- Consider the set of messages $\{\mu_{i \rightarrow j}(x_j)\}_{i,j}$ as a large state vector $\mu^t$ with $2|E(G)|r$ scalar elements.
- Each sent message moves the state vector from $\mu^t$ at time $t$ to $\mu^{t+1}$ at next time step.
- A parallel message moves the state vector as well.
- Convergence means that any set or subset of messages sent in parallel is such that $\mu^{t+1} = \mu^t$. 
Messages as matrix multiply

\[
\mu_{i \rightarrow j}(x_j) \propto \sum_{x_i} \psi_{i,j}(x_i, x_j) \psi_i(x_i) \prod_{k \in \delta(i) \setminus \{j\}} \mu_{k \rightarrow i}(x_i) \tag{7}
\]

\[
= \sum_{x_i} \psi'_{i,j}(x_i, x_j) \mu_{-j \rightarrow i}(x_i) \tag{8}
\]

\[
= (\psi'_{i,j})^T \mu_{-j \rightarrow i} \tag{9}
\]

- Here, \( \psi'_{i,j} \) is a matrix and \( \mu_{-j \rightarrow i} \) is a column vector.
- Going from state \( \mu^t \) to \( \mu^{t+1} \) is like matrix-vector multiply — group messages from \( \mu^t \) together into one vector representing \( \mu_{-j \rightarrow i} \) for each \( (i, j) \in E \), do the matrix-vector update, and store result in new state vector \( \mu^{t+1} \).
- If \( G \) is tree, \( \mu^t \) will converged after \( D \) steps.

Belief Propagation and Cycles

What if graph has cycles?

- MPP causes deadlock since there is no way to start sending messages
- Like before, we can assume that messages have an initial state, e.g., \( \mu_{i \rightarrow j}(x_j) = 1 \) for all \( (i, j) \in E(G) \) - note this is bi-directional. This breaks deadlock.
- We can then start sending messages. Will we converge after \( D \) steps? What does \( D \) even mean here?
- No, in fact we could oscillate forever.
Belief Propagation, Cycles, and Oscillation

- Consider odd length cycle (e.g., \(C_3, C_5\), etc.), \(C_3\) is sufficient \(i \rightarrow j \rightarrow k \rightarrow i\).
- Assume all messages start out at state \(\mu_{i \rightarrow j} = [1, 0]^T\).
- Consider (pairwise) edge functions, for each \(i, j\)

\[
\psi_{ij}(x_i, x_j) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{10}
\]

- then we have

\[
\mu_{j \rightarrow k}(x_k) = \sum_{x_j} \psi_{j,k}(x_j, x_k) \mu_{i \rightarrow j}(x_j) \tag{11}
\]

- or in matrix form

\[
\mu_{j \rightarrow k} = (\psi_{j,k})^T \mu_{i \rightarrow j} \tag{12}
\]

Let \(\mu^{t+1}_{i \rightarrow j}\) be the \(t^{th}\) formed message, with \(\mu^0_{i \rightarrow j}\) being the starting state at \([1, 0]^T\).
- Then \(\mu^1_{i \rightarrow j} = [0, 1]^T, \mu^2_{i \rightarrow j} = [1, 0]^T, \mu^3_{i \rightarrow j} = [0, 1]^T\), and so on, never converging. In fact,

\[
\mu^{t+1}_{i \rightarrow j} = (\psi_{i,j})^T \mu^t_{k \rightarrow i} \tag{13}
\]

\[
= (\psi_{i,j})^T (\psi_{k,i})^T \mu^t_{j \rightarrow k} \tag{14}
\]

\[
= (\psi_{i,j})^T (\psi_{k,i})^T (\psi_{j,k})^T \mu^t_{i \rightarrow j} \tag{15}
\]

\[
= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}^T \mu^t_{i \rightarrow j} \tag{16}
\]
Belief Propagation, Cycles, and Oscillation

- Thus, each time we go around the loop in the cycle, the message configuration for each \((i,j)\) will flip, thereby never converging.
- Damping the messages? I.e., Let \(0 \leq \gamma < 1\) and treat messages as

\[
\mu_{i \rightarrow j}^t \leftarrow \gamma \mu_{i \rightarrow j}^t + (1 - \gamma) \mu_{i \rightarrow j}^{t-1}
\]  

(18)

- Empirical Folklore - if we converge quickly without damping, the quality of the resulting marginals might be good. If we don’t converge quickly, w/o damping, might indicate some problem.
- Other message schedules
- Other forms of the interaction matrices
- Other initializations

If we initialize messages differently, things will turn out better.

- If \(\mu_{i \rightarrow j}^0 = [0.5, 0.5]^T\) then \(\mu_{i \rightarrow j}^{t+1} = \mu_{i \rightarrow j}^t\).
- Damping the messages appropriately will also end up at this configuration.
- Is there a better way to characterize this?
Belief Propagation, Single Cycle

- Consider a graph with a single cycle $C_\ell$.
- It could be a cycle with trees hanging off of each node. We send messages from the leaves of those dangling trees to the cycle (root) nodes, leaving only a cycle remaining.
- Consider what happens to $\mu_{i \rightarrow j}^t$ as $t$ increases. w.l.o.g. consider $\mu_{\ell \rightarrow 1}^t$
- Let the cycle be nodes (1, 2, 3, ... , $\ell$, 1)

\[
\mu_{\ell \rightarrow 1}^{t+1} = \left( \prod_{i=1}^{\ell-1} (\psi_{i,i+1})^T \right) \mu_{\ell \rightarrow 1}^t
\]

\[
= M \mu_{\ell \rightarrow 1}^t
\]

Will this converge to anything?

Theorem 1 (Power method lemma)

Let $A$ be a matrix with eigenvalues $\lambda_1, \ldots, \lambda_n$ (sorted in decreasing order) and corresponding eigenvectors $x_1, x_2, \ldots, x_n$. If $|\lambda_1| > |\lambda_2|$ (strict), then the update $x^{t+1} = \alpha A x^t$ converges to a multiple of $x_1$ starting from any initial vector $x^0 = \sum_i \beta_i x_i$ provided that $\beta_1 \neq 0$. The convergence rate factor is given by $|\lambda_2/\lambda_1|$. 
Belief Propagation, Single Cycle

From this, we the following theorem follows almost immediately.

**Theorem 2**

1. $\mu_{\ell \to 1}$ converges to the principle eigenvector of $M$.
2. $\mu_{2 \to 1}$ converges to the principle eigenvector of $M^T$.
3. The convergence rate is determined by the ratio of the largest and second largest eigenvalue of $M$.
4. The diagonal elements of $M$ correspond to correct marginal $p(x_1)$
5. The steady state “pseudo-marginal” $b(x_1)$ is related to the true marginal by $b(x_1) = \beta p(x_1) + (1 - \beta) q(x_1)$ where $\beta$ is the ratio of the largest eigenvalue of $M$ to the sum of all eigenvalues, and $q(x_1)$ depends on the eigenvectors of $M$.

**Proof.**

See Weiss2000.

What’s going on with our oscillating example?

- We had $M = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ which has row-eigenvector matrix $\begin{bmatrix} -1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$ with corresponding eigenvalues $-1$ and $1$.
- Note that any uniform vector will be “converged”, i.e., any vector of the form $[aa]$.
- However, we don’t have the *guaranteed* property of convergence since we don’t have that $|\lambda_1| > |\lambda_2|$. 
Belief Propagation, arbitrary graph

- This works for a graph with a single cycle, or a graph that contains a single cycle.
- It still does not tell us that we end up with correct marginals, rather we get “pseudo-marginals”, which are locally normalized, but might not be the correct marginals.
- Moreover, they might not be the correct marginals for any probability distribution.
- Also, we’d like a characterization of LBP’s convergence (if it happens) for more general graphs, with an arbitrary number of loops.

Graphical Models, Exponential Families, and Variational Inference

- We’re going to start covering our book.
- Today we will start on chapter 3 (we assume you will read chapters 1 and 2 on your own).
- We’ll follow the Wainwright and Jordan notation, will point out where it conflicts a bit with the current notation we’ve been using.
**exponential family models**

- \( \phi = (\phi_\alpha, \alpha \in \mathcal{I}) \) is a collection of functions known as potential functions or sufficient statistics, where \( \mathcal{I} \) is the index set.
- Each \( \phi_\alpha \) is a function of \( x \), \( \phi_\alpha(x) \) but it usually does not use all of \( x \) (only a subset of elements).
- \( \theta \) is a vector of **canonical parameters** (same length, \( |\mathcal{I}| \)). \( \theta \in \Omega \subseteq \mathbb{R}^d \) where \( d = |\mathcal{I}| \).
- We can define a family as
  \[
  p_\theta(x) = \exp(\langle \theta, \phi(x) \rangle - A(\theta)) \quad (21)
  \]
  Note that we’re using \( \phi \) here in the exponent, before we were using it out of the exponent.
- Note that \( \phi(x) = (\phi_1(x), \phi_2(x), \ldots, \phi_{|\mathcal{I}|}) \) where again each \( \phi_i(x) \) might use only some of the elements in vector \( x \).

**exponential family models and clique features**

- Given a graph \( G = (V, E) \) we have a set of cliques \( C \).
- In order to respect the graph, we have to make sure that \( \alpha \in \mathcal{I} \)
  respects the cliques. That is, for any \( \alpha \in \mathcal{I} \), and feature function \( \phi_\alpha(x_\alpha) \) there must be a clique \( C \in C \) such that \( \alpha \subseteq C \).
- On the other hand, by having a different index set \( \mathcal{I} \) we can have more than one feature (sufficient statistic) for a given clique. That is, for any given \( C \in C \) we might have multiple \( \alpha \) such that \( \alpha \subseteq C \) for some clique \( C \in C \).
- Example: simple discrete random variable \( X \in \{1, 2, \ldots, k\} \) might have
  \( \alpha_i(x) \triangleq 1(x = i) \).
- In the past, \( \alpha \) are the cliques of some graph (still the case here), but not nec. maxcliques. Not nec. dealing with triangulated models. Could be based on cliques or subsets of cliques.
- Key thing: \( p \in \mathcal{F}(G, R(f)) \) by Hammersley-Clifford theorem, where \( G \) is formed by using \( \alpha \) as an edge clique cover (i.e., if there is an \( \alpha \in \mathcal{I} \)
  such that \( u, v \in \alpha \) where \( u, v \in V(G) \), then there must be an edge \( (u, v) \in E(G) \).
exponential family models

- exponential models are in our sense sufficient to deal with the computational aspects graphical models.
- We can have \( p \in \mathcal{F}((V, E), R^{(f)}) \) implies \( p \in \mathcal{F}((V, E + E_1), R^{(f)}) \) but in some sense, for any \( G \), we want to deal with the models for which \( G \) is tight (we don’t want to use overly complex graph to deal with family that is simpler).
- Exponential models can represent any factorization, given any factorization in terms of \( \phi \), we can do \( \exp(\log \phi) \) to get potentials.
- We can often make them log-linear models as well with the right potential functions which won’t increase tree-width of the graph.
- Moreover, exponential family models are incredibly flexible and have a number of desirable properties (e.g., aspects of the log partition function which we will see)

Underlying base measure \( \nu \), \( p \) is absolutely continuous w.r.t. \( \nu \)

A measure \( \nu \) is absolutely continuous with respect to \( \mu \) if for each \( A \in \mathcal{F}, \mu(A) = 0 \) implies \( \nu(A) = 0 \). In this case \( \nu \) is also said to be dominated by \( \mu \) (if \( \mu \) goes to zero, so must \( \nu \)), and the relation is indicated by \( \nu \ll \mu \). If \( \nu \ll \mu \) and \( \mu \ll \nu \), the measures are equivalent, indicated by \( \nu \equiv \nu \).

Based on underlying set of parameters \( \theta \), we have family of models

\[
p_{\theta}(x) = \frac{1}{Z(\theta)} \exp \left\{ \sum_{\alpha \in \mathcal{I}} \theta_{\alpha} \phi_{\alpha}(x) \right\} = \exp(\langle \theta, \phi(x) \rangle - A(\theta)) \quad (22)
\]

family can arise for a number of reasons, e.g., distribution having maximum entropy but that satisfies certain (moment) constraints.

Given data \( D = \{\bar{x}_E^{(i)}\}_{i=1}^M \), form the expected statistics (requirements) of a model

\[
\hat{\mu}_{\alpha} = \frac{1}{M} \sum_{i=1}^M \phi_{\alpha}(\bar{x}^{(i)}) \quad (23)
\]
Exponential family models

- Goal is to find

\[ p^* \in \text{argmax } H(p) \text{ s.t. } \mathbb{E}_p[\phi_\alpha(X)] = \hat{\mu}_\alpha \quad \forall \alpha \in \mathcal{I} \quad (24) \]

where \( \forall \alpha \in \mathcal{I} \)

\[ \mathbb{E}_p[\phi_\alpha(X)] = \int_{\mathcal{D}_X} \phi_\alpha(x)p(x)\nu(dx) \quad (25) \]

- This is solved by taking a distribution in the form of Eq. 22, by finding \( \theta \) that solves

\[ E_{\theta}[\phi_\alpha(X)] = \hat{\mu}_\alpha \text{ for all } \alpha \in \mathcal{I} \quad (26) \]

- To ensure normalized, we use log partition (cumulant) function

\[ A(\theta) = \log \int_{\mathcal{D}_X} \exp(\langle \theta, \phi(x) \rangle) \nu(dx) \quad (27) \]

with \( \theta \in \Omega \triangleq \{ \theta \in \mathbb{R}^d \mid A(\theta) < +\infty \} \)

- \( A(\theta) \) is convex function of \( \theta \)

- Exponential family for which \( \Omega \) is open is called regular
Minimal Representation of Exponential Family

\[ p_\theta(x) = \exp(\langle \theta, \phi(x) \rangle - A(\theta)) \]  
(28)

where \( A(\theta) = \log \int_{D_X} \exp(\langle \theta, \phi(x) \rangle) \nu(dx) \)  
(29)

- Minimal representation - Does not exist a nonzero vector \( \gamma \in \mathbb{R}^d \) for which \( \langle \gamma, \phi(x) \rangle \) is constant \( \forall x \) that are \( \nu \)-measurable.

- Essential idea: that for a set of sufficient stats \( I \), there shouldn’t be any lower-dim vector \( |I'| < |I| \) that is also sufficient (a min suf stat is a function of all other suf stats). We can’t reduce the dimensionality \( d \) without reducing the family.

Overcomplete Representation

\[ p_\theta(x) = \exp(\langle \theta, \phi(x) \rangle - A(\theta)) \]  
(30)

where \( A(\theta) = \log \int_{D_X} \exp(\langle \theta, \phi(x) \rangle) \nu(dx) \)  
(31)

- Overcomplete representation \( d \) higher than need be
- Exists affine hyperplane of different parameters that induce exactly same distribution
- I.e., \( \exists a \text{ s.t. } \langle a, \phi(x) \rangle = c \ \forall x \text{ where } c = \text{ constant} \).
- We’ll see later, this useful in understanding BP algorithm.
Exponential family models

- Minimal representation of Bernoulli distribution is
  \[ p(x|\gamma) = \exp(\gamma x - A(\gamma)) \]  
  (32)

- Overcomplete rep of Bernoulli dist.
  \[ p(x|\theta_0, \theta_1) = \exp(\langle \theta, \phi(x) \rangle) \]  
  \[ = \exp(\theta_0(1 - x) + \theta_1 x - A(\gamma)) \]  
  (33)

  \[ = \exp(\theta_0(1 - x) + \theta_1 x - A(\gamma)) \]  
  (34)

  where \( \theta = (\theta_0, \theta_1) \) and \( \phi(x) = (1 - x, x) \).

- Is there a vector \( a \) s.t. \( \langle a, \phi(x) \rangle = c \) for all \( x \), \( \nu \)-a.e.?

- If \( a = (1, 1) \) then \( \langle a, \phi(x) \rangle = (1 - x) + x = 1 \)

- In general, this is overcomplete since there is a linear combination of feature functions that are constant. Any parameters of form \( \theta_1 - \theta_0 = \gamma \) give same distribution.

Famous Example - Ising Model

- Famous example is the Ising model in statistical physics. We have a grid network with pairwise interactions, each variable is 0/1-valued binary, and parameters associated with pairs being both on. Model becomes

  \[ p_\theta(x) = \exp \left\{ \sum_{v \in V} \theta_v x_v + \sum_{(s,t) \in E} \theta_{st} x_s x_t - A(\theta) \right\}, \]  
  (35)

  with

  \[ A(\theta) = \log \sum_{x \in \{0,1\}^m} \exp \left\{ \sum_{v \in V} \theta_v x_v + \sum_{(s,t) \in E} \theta_{st} x_s x_t - A(\theta) \right\} \]  
  (36)

- Note that this is in minimal form. Any change to parameters will result in different distribution.
Multivalued variables

- Variables need not binary, instead let \( D_X = \{0, 1, \ldots, r - 1\} \) for \( r > 2 \).
- We can define a set of indicator functions constituting minimal sufficient statistics. That is

\[
1_{s,j}(x_s) = \begin{cases} 
1 & \text{if } x_s = j \\
0 & \text{else}
\end{cases}
\]  

and

\[
1_{st;jk}(x_s, x_t) = \begin{cases} 
1 & \text{if } x_s = j \text{ and } x_t = k, \\
0 & \text{else}
\end{cases}
\]

- Model becomes

\[
p_\theta(x) = \exp \left\{ \sum_{v \in V} \sum_{i=0}^{r-1} \theta_{v;j} 1_{s;j}(x_v) + \sum_{(s,t) \in E} \sum_{j,k} \theta_{st;jk} 1_{st;jk}(x_s, x_t) - A(\theta) \right\},
\]

This is overcomplete, why?

Multivariate Gaussian

- Usually, multivariate Gaussian is parameterized via mean and covariance matrix. For canonical exponential form, we use mean and correlation matrix. I.e.

\[
p_\theta(x) = \exp \left\{ \langle \theta, x \rangle + \frac{1}{2} \langle \Theta, xx^T \rangle - A(\theta, \Theta) \right\}
\]

- So sufficient statistics are \((x_i)_{i=1}^n\) and \((x_i x_j)_{i,j}\)
- Mixtures of Gaussians can also be parameterized in exponential form (but note, key is that it is the joint distribution \( p_{\theta_s}(y_s, x_s) \)).
Other examples

A few other examples in the book

- Mixture models
- Latent Dirichlet Allocation, and general hierarchical Bayesian models. Key here is that it is for one expansion, not variable.
- Models with hard constraints - key thing is to place the hard constraints in the \( \nu \) measure. Sufficient statistics become easy if complexity is encoded in the measure. Alternative is to allow features over extended reals (i.e., a feature can provide \(-\infty\) but this leads to certain technical difficulties that they would rather not deal with).

Mean Parameters, Convex Cores

- Consider quantities \( \mu_\alpha \) associated with statistic \( \phi_\alpha \) defined as:

\[
\mu_\alpha = \mathbb{E}_p[\phi_\alpha(X)] = \int_{\mathcal{D}_X} \phi_\alpha(x)p(x)\nu(dx) \quad (41)
\]

- this defines a vector of “mean parameters” \((\mu_1, \mu_2, \ldots, \mu_d)\) with \( d = |\mathcal{I}| \).
- Define all the possible such vectors

\[
\mathcal{M} \triangleq \left\{ \mu \in \mathbb{R}^d : \exists p \text{ s.t. } \mu_\alpha = \mathbb{E}_p[\phi_\alpha(X)] \forall \alpha \in \mathcal{I} \right\} \quad (42)
\]

- didn’t say \( p \) was exponential family
- \( \mathcal{M} \) is convex since expected value is a linear operator. So convex combinations of \( p \) and \( p' \) will lead to convex combinations of \( \mu \) and \( \mu' \)
- \( \mathcal{M} \) is like a “convex core” of all distributions expressed via \( \phi \).
Mean Parameters and Gaussians

- Here, we have $\mathbb{E}[XX^T] = C$ and $\mu = \mathbb{E}X$. Question is, how to define $\mathcal{M}$?
- Given definition of $C$ and $\mu$, then $C - \mu\mu^T$ must be valid covariance matrix (since this is $\mathbb{E}[X - \mathbb{E}X]$).
- Thus, $C - \mu\mu^T \succeq 0$, thus p.s.d. matrix.
- On the other hand, if this is true, we can form a Gaussian using $C - \mu\mu^T$ as the covariance matrix.
- Thus, for Gaussian MRFs, $\mathcal{M}$ has the form

$$\mathcal{M} = \left\{ (\mu, C) \in \mathbb{R}^m \times \mathcal{S}^m_+ | C - \mu\mu^T \succeq 0 \right\}$$  \hspace{1cm} (43)

where $\mathcal{S}^m_+$ is the set of symmetric positive semi-definite matrices.

Mean Parameters and Polytopes

- When $X$ is discrete, we get a polytope since

$$\mathcal{M} = \left\{ \mu \in \mathbb{R}^b : \mu = \sum_x \phi(x)p(x) \text{ for some } p \in \mathcal{U} \right\}$$  \hspace{1cm} (44)

$$= \text{convexhull}\{\phi(x), x \in \mathcal{D}_X\}$$  \hspace{1cm} (45)

- So we have a polytope
Mean Parameters and Polytopes

- Polytopes can be represented as a set of linear inequalities, i.e., there is a $|J| \times d$ matrix $A$ and $|J|$-element column vector $b$ with

$$M = \left\{ \mu \in \mathbb{R}^d : A\mu \geq b \right\} = \left\{ \mu \in \mathbb{R}^d : \langle a_j, \mu \rangle \geq b_j, \forall j \in J \right\}$$

(46)

with $A$ having rows $a_j$.

Example: Ising mean parameters

$$\mu_v = \mathbb{E}_p[X_v] = p(X_v = 1) \quad \forall v \in V$$

(47)

$$\mu_{s,t} = \mathbb{E}_p[X_s X_t] = p(X_s = 1, X_t = 1) \quad \forall (s, t) \in E(G)$$

(48)

- In this case, the mean parameters lie in a polytope that represent the probabilities of a node being 1 or a pair of adjacent nodes being 1, 1 for each node and edge in the graph.
Mean Parameters and Overcomplete Representation

- We can use overcomplete representation and get a “marginal polytope”, a polytope that represents the marginal distributions at each potential function.
- Example: Ising overcomplete potential functions (generalization of Bernoulli example we saw before)

\[
\forall v \in V(G), j \in \{1 \ldots r\}, \text{ define } \phi_{v,j}(x_v) = 1(x_v = j) \tag{49}
\]

\[
\forall (s, t) \in E(G), j, k \in \{1 \ldots r\}, \text{ define } \phi_{st,jk}(x_s, x_t) = 1(x_s = j, x_t = k) \tag{51}
\]

- So we now have \(|V|r + 2|E|r^2\) functions each with a corresponding parameter.

Mean Parameters and Marginal Polytopes

- Mean parameters are now true marginals, i.e., \(\mu_v(j) = p(x_v = j)\) and \(\mu_{st}(j, k) = p(x_s = j, x_t = k)\) since

\[
\mu_{v,j} = \mathbb{E}_p[1(x_v = j)] = p(x_v = j) \tag{52}
\]

\[
\mu_{st,jk} = \mathbb{E}_p[1(x_s = j, x_t = k)] = p(x_s = j, x_t = k) \tag{53}
\]

- Such an \(\mathcal{M}\) is called the marginal polytope. Any \(\mu\) must live in the polytope that corresponds to node and edge true marginals!!

- This polytope can help us to characterize when BP converges (there might be an outer bound of this polytope), or it might characterize the result of a mean-field approximation (an inner bound of this polytope) as we’ll see.
Learning is the dual of Inference

- We can view the inference problem as moving from the canonical parameters $\theta$ to the point in the marginal polytope, called *forward mapping*, moving from $\theta \in \Omega$ to $\mu \in \mathcal{M}$.

- We can view the (maximum likelihood) learning problem as moving from a point in the polytope (empirical distribution) to the canonical parameters.

- Graph structure (e.g., tree-width) makes this easy or hard, and also characterizes the polytope (how complex it is in terms of number of faces).

Ex: Estimate $\theta$ with $\hat{\theta}$ based on data $D = \{\bar{x}_E^{(i)}\}_{i=1}^M$ of size $M$, likelihood function

$$
\ell(\theta, D) = \frac{1}{M} \sum_{i=1}^M \log p_\theta(\bar{x}^{(i)}) = \langle \theta, \hat{\mu} \rangle - A(\theta)
$$

(54)

where

$$
\hat{\mu} = \hat{E}[\phi(X)] = \frac{1}{M} \sum_{i=1}^M \phi(\bar{x}^{(i)})
$$

(55)

By taking derivatives of the above, it is easy to see that solution is point $\theta$ such that

$$
\mathbb{E}_\theta[\phi(X)] = \hat{\mu}
$$

(56)

This is the the *backward mapping problem*, going from $\mu$ to $\theta$. 
Learning is the dual of Inference

- In other words, the solution to the maximum likelihood problem is one that satisfies the moment constraints and has the exponential model form. The exponential model form is exactly the equation that arises when we find the maximum entropy distribution over those distributions satisfying the moment constraints.

- This shows that maximum entropy learning under a set of constraints (given by $\mathbb{E}_\theta[\phi(X)] = \hat{\mu}$) is the same as maximum likelihood learning of an exponential model form.

- If we do maximum entropy learning, where does the exp pop up? From the entropy function. I.e., the exponential form is the distribution that has maximum entropy having those constraints.

Log partition function

\[ A(\theta) = \log \int_{D_X} \langle \theta, \phi(x) \rangle \nu(dx) \] (57)

- If we know the log partition function, we know a lot for an exponential family model. In particular, we know
  - $A(\theta)$ is convex in $\theta$
  - It yields cumulants of the random vector $\phi(X)$

\[ \frac{\partial A}{\partial \theta_\alpha}(\theta) = \mathbb{E}_\theta[\phi_\alpha(X)] = \int_{D_X} \phi_\alpha(X) p_\theta(x) \nu(dx) \] (58)

\[ \frac{\partial^2 A}{\partial \theta_{\alpha_1} \partial \theta_{\alpha_2}}(\theta) = \mathbb{E}_\theta[\phi_{\alpha_1}(X)\phi_{\alpha_2}(X)] - \mathbb{E}_\theta[\phi_{\alpha_1}(X)] \mathbb{E}_\theta[\phi_{\alpha_2}(X)] \] (59)
Log partition function

- So derivative of log partition function w.r.t. $\theta$ is equal to our mean parameter $\mu$ in the discrete case.
- Given $A(\theta)$, we can recover the marginals for each potential function $\phi_\alpha, \alpha \in \mathcal{I}$ (when mean parameters lie in the marginal polytope).
- If we can approximate $A(\theta)$ with $\tilde{A}(\theta)$ then we can get approximate marginals. Perhaps we can bound it without inordinate compute resources.
- The Bethe approximation (as we’ll see) is such an approximation and corresponds to fixed points of loopy belief propagation.
- In some rarer cases, we can bound the approximation (current research trend).

\[
\nabla A : \Omega \rightarrow \mathcal{M}' \text{ where } \mathcal{M}' \subseteq \mathcal{M},
\]
for minimal exponential family models, this mapping is one-to-one, that is $\mathcal{M}' = \mathcal{M}$ and there is a unique pairing between $\mu$ and $\theta$.
- For non-minimal exponential families, we have have more than one $\theta$ (not surprising since multiple $\theta$'s can yield the same distribution).
- For non-exponential families, other distributions can yield $\mu$, but the exponential family one is the one that has maximum entropy (consider a Gaussian, which is the distribution having a given mean and covariance having maximum entropy amongst all other distributions).
- Key point: all mean parameters are realizable by member of exp. family.
- There exists some technical points about interior of $\mathcal{M}$ we won’t discuss here.
Conjugate Duality

- Maximum likelihood problem for exp. family
  \[ \theta^* \in \arg\max_{\theta} (\langle \theta, \hat{\mu} \rangle - A(\theta)) \] (60)

- Conjugate dual
  \[ A^*(\mu) \triangleq \sup_{\theta \in \Omega} (\langle \theta, \mu \rangle - A(\theta)) \] (61)

  dual is like ML when \( \mu \in M \)

  Key: when \( \mu \in M \), dual is negative entropy of exp. model \( p_{\theta(\mu)} \) where \( \theta(\mu) \) is the unique set of canonical parameters satisfying this matching condition

  \[ \mathbb{E}_{\theta(\mu)}[\phi(X)] = \nabla A(\theta(\mu)) = \mu \] (62)

  When \( \mu \notin M \), then \( A^*(\mu) = +\infty \), so dual optimization need consider points only in \( M \).

Theorem 3

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

\[
A^*(\mu) = \begin{cases} 
-H(p_{\theta(\mu)}) & \text{if } \mu \in M \\
+\infty & \text{otherwise}
\end{cases}
\] (63)

(b) Partition function has variational representation

\[ A(\theta) = \sup_{\mu \in M} \{ \langle \theta, \mu \rangle - A^*(\mu) \} \] (64)

(c) For \( \theta \in \Omega \), sup occurs at \( \mu \in M \) at matching conditions

\[ \mu = \int_{D_X} \phi(x)p_{\theta}(x)\nu(dx) = \mathbb{E}_\theta[\phi(X)] = \nabla A(\theta) \] (65)
Conjugate Duality

- Note that $A^*$ isn’t exactly entropy, but is only entropy sometimes.
- $A(\theta)$ in previous expression is the “inference” problem (dual of the dual) for a given $\theta$, whenever $\mu \notin \mathcal{M}$ we’ve got $-\infty$ which can’t be sup, so need only consider $\mathcal{M}$.
- Computing $A(\theta)$ in this way corresponds to the inference problem (finding mean parameters, or node and edge marginals). Key: we compute the log partition function simultaneously with solving inference, given the dual.
- Good news: problem is concave objective over a convex set. Should be easy. In simple examples, indeed, it is easy.
- Bad news: $\mathcal{M}$ is quite complicated to characterize, depends on the complexity of the graphical model.
- More bad news: $A^*$ not given explicitly in general and hard to compute.

Some good news: The above form gives us new avenues to do approximation.

Surprisingly, this is strongly related to belief propagation (i.e., the sum-product commutative semiring)!!
<table>
<thead>
<tr>
<th>Logistics</th>
<th>Review</th>
<th>LBP</th>
<th>Next phase of class</th>
<th>exp models</th>
<th>Scratch</th>
<th>Summary</th>
</tr>
</thead>
</table>

**Scratch Paper**
Sources for Today’s Lecture

- Most of this material comes from the Wainwright and Jordan book.