Approximate marginal inference

- Given the joint $p(x_1, \ldots, x_n)$ represented as a graphical model, how do we perform **marginal inference**, e.g. to compute $p(x_1 \mid e)$?
- We showed in Lecture 4 that doing this exactly is NP-hard
- Nearly all **approximate inference** algorithms are either:
  1. Monte-carlo methods (e.g., Gibbs sampling, likelihood reweighting, MCMC)
  2. Variational algorithms (e.g., mean-field, loopy belief propagation)
Variational methods

- **Goal**: Approximate difficult distribution $p(x \mid e)$ with a new distribution $q(x)$ such that:
  1. $p(x \mid e)$ and $q(x)$ are “close”
  2. Computation on $q(x)$ is easy

- How should we measure distance between distributions?
  - The **Kullback-Leibler divergence** (KL-divergence) between two distributions $p$ and $q$ is defined as
    
    $$ D(p \parallel q) = \sum_x p(x) \log \frac{p(x)}{q(x)} $$

    (measures the expected number of extra bits required to describe *samples from* $p(x)$ using a code based on $q$ instead of $p$)

    - $D(p \parallel q) \geq 0$ for all $p, q$, with equality if and only if $p = q$
    - Notice that KL-divergence is **asymmetric**
KL-divergence \((\text{see Section 2.8.2 of Murphy})\)

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

Suppose \(p\) is the true distribution we wish to do inference with.

What is the difference between the solution to

\[
\arg\min_q D(p\|q)
\]

(called the \(M\)-projection of \(q\) onto \(p\)) and

\[
\arg\min_q D(q\|p)
\]

(called the \(I\)-projection)?

These two will differ only when \(q\) is minimized over a restricted set of probability distributions \(Q = \{q_1, \ldots\}\), and in particular when \(p \notin Q\).
KL-divergence – M-projection

\[ q^* = \arg \min_{q \in Q} D(p \| q) = \sum_x p(x) \log \frac{p(x)}{q(x)}. \]

For example, suppose that \( p(z) \) is a 2D Gaussian and \( Q \) is the set of all Gaussian distributions with diagonal covariance matrices:

\[ p=\text{Green}, \quad q^*=\text{Red} \]
KL-divergence – I-projection

\[ q^* = \arg \min_{q \in Q} D(q \parallel p) = \sum_x q(x) \log \frac{q(x)}{p(x)}. \]

For example, suppose that \( p(z) \) is a 2D Gaussian and \( Q \) is the set of all Gaussian distributions with diagonal covariance matrices:

\[ p=\text{Green, } q^*=\text{Red} \]
In this simple example, both the M-projection and I-projection find an approximate $q(x)$ that has the correct mean (i.e. $E_p[z] = E_q[z]$):

What if $p(x)$ is multi-modal?
KL-divergence – M-projection (mixture of Gaussians)

\[ q^* = \arg \min_{q \in Q} D(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)}. \]

Now suppose that \( p(x) \) is mixture of two 2D Gaussians and \( Q \) is the set of all 2D Gaussian distributions (with arbitrary covariance matrices):

\[ p=\text{Blue}, \quad q^*=\text{Red} \]

M-projection yields distribution \( q(x) \) with the correct mean and covariance.
KL-divergence – I-projection (mixture of Gaussians)

\[ q^* = \arg \min_{q \in Q} D(q\|p) = \sum_x q(x) \log \frac{q(x)}{p(x)}. \]

\( p=\text{Blue}, \ q^*=\text{Red} \) (two local minima!)

Unlike M-projection, the I-projection does not always yield the correct moments.

Q: \( D(p\|q) \) is convex – so why are there local minima?
A: using a parametric form for \( q \) (i.e., a Gaussian). Not convex in \( \mu, \Sigma \).
Recall that the M-projection is:

\[ q^* = \arg \min_{q \in Q} D(p \| q) = \sum_x p(x) \log \frac{p(x)}{q(x)}. \]

Suppose that \( Q \) is an exponential family (\( p(x) \) can be arbitrary) and that we perform the M-projection, finding \( q^* \)

**Theorem:** The expected sufficient statistics, with respect to \( q^*(x) \), are exactly the marginals of \( p(x) \):

\[ E_{q^*}[f(x)] = E_p[f(x)] \]

Thus, solving for the M-projection (exactly) is just as hard as the original inference problem.
M-projection does moment matching

- Recall that the M-projection is:

\[ q^* = \arg \min_{q(x; \eta) \in Q} D(p\|q) = \sum_x p(x) \log \frac{p(x)}{q(x)}. \]

- **Theorem:** \( E_{q^*}[f(x)] = E_p[f(x)] \).

- **Proof:** Look at the first-order optimality conditions.

\[
\begin{align*}
\partial_{\eta_i} D(p\|q) &= -\partial_{\eta_i} \sum_x p(x) \log q(x) \\
&= -\partial_{\eta_i} \sum_x p(x) \log \left\{ h(x) \exp\{\eta \cdot f(x) - \ln Z(\eta)\} \right\} \\
&= -\partial_{\eta_i} \sum_x p(x) \left\{ \eta \cdot f(x) - \ln Z(\eta) \right\} \\
&= -\sum_x p(x)f_i(x) + E_{q(x;\eta)}[f_i(x)] \quad \text{(since } \partial_{\eta_i} \ln Z(\eta) = E_q[f_i(x)] \text{)} \\
&= -E_p[f_i(x)] + E_{q(x;\eta)}[f_i(x)] = 0.
\end{align*}
\]

- **Corollary:** Even computing the gradients is hard (can’t do gradient descent)
Most variational inference algorithms make use of the l-projection
Variational methods

- Suppose that we have an arbitrary graphical model:

\[ p(x; \theta) = \frac{1}{Z(\theta)} \prod_{c \in C} \phi_c(x_c) = \exp \left( \sum_{c \in C} \theta_c(x_c) - \ln Z(\theta) \right) \]

- All of the approaches begin as follows:

\[
D(q\|p) = \sum_x q(x) \ln \frac{q(x)}{p(x)}
\]

\[
= - \sum_x q(x) \ln p(x) - \sum_x q(x) \ln \frac{1}{q(x)}
\]

\[
= - \sum_x q(x) \left( \sum_{c \in C} \theta_c(x_c) - \ln Z(\theta) \right) - H(q(x))
\]

\[
= - \sum_{c \in C} \sum_x q(x) \theta_c(x_c) + \sum_x q(x) \ln Z(\theta) - H(q(x))
\]

\[
= - \sum_{c \in C} E_q[\theta_c(x_c)] + \ln Z(\theta) - H(q(x)).
\]
Mean field algorithms for variational inference

$$\max_{q \in Q} \sum_{c \in C} \sum_{x_c} q(x_c) \theta_c(x_c) + H(q(x)).$$

- Although this function is concave and thus in theory should be easy to optimize, we need some compact way of representing $q(x)$.
- *Mean field* algorithms assume a factored representation of the joint distribution, e.g.

$$q(x) = \prod_{i \in V} q_i(x_i) \quad \text{(called *naive* mean field)}$$
Naive mean-field

Suppose that $Q$ consists of all fully factored distributions, of the form

\[ q(x) = \prod_{i \in V} q_i(x_i) \]

We can use this to simplify

\[
\max_{q \in Q} \sum_{c \in C} \sum_{x_c} q(x_c) \theta_c(x_c) + H(q)
\]

First, note that $q(x_c) = \prod_{i \in c} q_i(x_i)$

Next, notice that the joint entropy decomposes as a sum of local entropies:

\[
H(q) = -\sum_x q(x) \ln q(x)
\]

\[
= -\sum_x q(x) \ln \prod_{i \in V} q_i(x_i) = -\sum_x q(x) \sum_{i \in V} \ln q_i(x_i)
\]

\[
= -\sum_{i \in V} \sum_x q(x) \ln q_i(x_i)
\]

\[
= -\sum_{i \in V} \sum_{x_i} q_i(x_i) \ln q_i(x_i) \sum_{x_{V\setminus i} | x_i} q(x_{V\setminus i} | x_i) = \sum_{i \in V} H(q_i).
\]
Suppose that $Q$ consists of all fully factored distributions, of the form $q(x) = \prod_{i \in V} q_i(x_i)$.

We can use this to simplify

$$
\max_{q \in Q} \sum_{c \in C} \sum_{x_c} q(x_c) \theta_c(x_c) + H(q)
$$

First, note that $q(x_c) = \prod_{i \in c} q_i(x_i)$.

Next, notice that the joint entropy decomposes as $H(q) = \sum_{i \in V} H(q_i)$.

Putting these together, we obtain the following variational objective:

$$(\ast) \max_{q} \sum_{c \in C} \sum_{x_c} \theta_c(x_c) \prod_{i \in c} q_i(x_i) + \sum_{i \in V} H(q_i)$$

subject to the constraints

$$q_i(x_i) \geq 0 \quad \forall i \in V, x_i \in \text{Val}(X_i)$$

$$\sum_{x_i \in \text{Val}(X_i)} q_i(x_i) = 1 \quad \forall i \in V$$
Naive mean-field for pairwise MRFs

- How do we maximize the variational objective?

\[
(*) \max_q \sum_{ij \in E} \sum_{x_i, x_j} \theta_{ij}(x_i, x_j) q_i(x_i) q_j(x_j) - \sum_{i \in V} \sum_{x_i} q_i(x_i) \ln q_i(x_i)
\]

- This is a non-concave optimization problem, with many local maxima!

- Nonetheless, we can greedily maximize it using **block coordinate ascent**:
  1. Iterate over each of the variables \(i \in V\). For variable \(i\),
  2. Fully maximize (*) with respect to \(\{q_i(x_i), \forall x_i \in \text{Val}(X_i)\}\).
  3. Repeat until convergence.

- Constructing the Lagrangian, taking the derivative, setting to zero, and solving yields the update: *(shown on blackboard)*

\[
q_i(x_i) \leftarrow \frac{1}{Z_i} \exp \left\{ \theta_i(x_i) + \sum_{j \in N(i)} \sum_{x_j} q_j(x_j) \theta_{ij}(x_i, x_j) \right\}
\]
How accurate will the approximation be?

- Consider a distribution which is an XOR of two binary variables $A$ and $B$: $p(a, b) = 0.5 - \epsilon$ if $a \neq b$ and $p(a, b) = \epsilon$ if $a = b$

- The contour plot of the variational objective is:

- Even for a single edge, mean field can give very wrong answers!
- Interestingly, once $\epsilon > 0.1$, mean field has a single maximum point at the uniform distribution (thus, exact)
Structured mean-field approximations

- Rather than assuming a fully-factored distribution for $q$, we can use a \textit{structured} approximation, such as a spanning tree.

- For example, for a factorial HMM, a good approximation may be a product of chain-structured models:

\[ g_{\beta}(\mu(F)) = \mu_s \mu_t \mu_u. \]
Recall our starting place for variational methods...

Suppose that we have an arbitrary graphical model:

\[
p(x; \theta) = \frac{1}{Z(\theta)} \prod_{c \in C} \phi_c(x_c) = \exp \left( \sum_{c \in C} \theta_c(x_c) - \ln Z(\theta) \right)
\]

All of the approaches begin as follows:

\[
D(q\|p) = \sum_x q(x) \ln \frac{q(x)}{p(x)}
\]

\[
= - \sum_x q(x) \ln p(x) - \sum_x q(x) \ln \frac{1}{q(x)}
\]

\[
= - \sum_x q(x) \left( \sum_{c \in C} \theta_c(x_c) - \ln Z(\theta) \right) - H(q(x))
\]

\[
= - \sum_{c \in C} \sum_x q(x) \theta_c(x_c) + \sum_x q(x) \ln Z(\theta) - H(q(x))
\]

\[
= - \sum_{c \in C} E_q[\theta_c(x_c)] + \ln Z(\theta) - H(q(x)).
\]
The log-partition function

- Since $D(q\|p) \geq 0$, we have
  \[
  -\sum_{c \in C} E_q[\theta_c(x_c)] + \ln Z(\theta) - H(q(x)) \geq 0,
  \]
  which implies that
  \[
  \ln Z(\theta) \geq \sum_{c \in C} E_q[\theta_c(x_c)] + H(q(x)).
  \]

- Thus, any approximating distribution $q(x)$ gives a lower bound on the log-partition function (for a BN, this is the log probability of the observed variables)

- Recall that $D(q\|p) = 0$ if and only if $p = q$. Thus, if we allow ourselves to optimize over all distributions, we have:
  \[
  \ln Z(\theta) = \max_q \sum_{c \in C} E_q[\theta_c(x_c)] + H(q(x)).
  \]
Re-writing objective in terms of moments

\[
\ln Z(\theta) = \max_q \sum_{c \in C} E_q[\theta_c(x_c)] + H(q(x)) \\
= \max_q \sum_{c \in C} \sum_{x} q(x) \theta_c(x_c) + H(q(x)) \\
= \max_q \sum_{c \in C} \sum_{x_c} q(x_c) \theta_c(x_c) + H(q(x)).
\]

- Now assume that \(p(x)\) is in the exponential family, and let \(f(x)\) be its sufficient statistic vector
- Define \(\mu_q = E_q[f(x)]\) to be the marginals of \(q(x)\)
- We can re-write the objective as

\[
\ln Z(\theta) = \max_{\mu \in M} \max_{q: E_q[f(x)] = \mu} \sum_{c \in C} \sum_{x_c} \theta_c(x_c) \mu_c(x_c) + H(q(x)),
\]

where \(M\), the marginal polytope, consists of all valid marginal vectors
Re-writing objective in terms of moments

Next, push the max over $q$ instead to obtain:

$$\ln Z(\theta) = \max_{\mu \in M} \sum_{c \in C} \sum_{x_c} \theta_c(x_c) \mu_c(x_c) + H(\mu),$$

where

$$H(\mu) = \max_{q:E_q[f(x)] = \mu} H(q) \quad \leftarrow \text{Does this look familiar?}$$

For discrete random variables, the **marginal polytope** $M$ is given by

$$M = \left\{ \mu \in \mathbb{R}^d \mid \mu = \sum_{x \in \mathcal{X}^m} p(x)f(x) \text{ for some } p(x) \geq 0, \sum_{x \in \mathcal{X}^m} p(x) = 1 \right\}$$

$$= \text{conv}\left\{ f(x), x \in \mathcal{X}^m \right\} \quad (\text{conv denotes the convex hull operation})$$

For a discrete-variable MRF, the sufficient statistic vector $f(x)$ is simply the concatenation of indicator functions for each clique of variables that appear together in a potential function.

For example, if we have a pairwise MRF on binary variables with $m = |V|$ variables and $|E|$ edges, $d = 2m + 4|E|$.
Marginal polytope for discrete MRFs

Marginal polytope

(Wainwright & Jordan, '03)

valid marginal probabilities

Assignment for $X_1$

Assignment for $X_2$

Assignment for $X_3$

Edge assignment for $X_1X_3$

Edge assignment for $X_1X_2$

Edge assignment for $X_2X_3$

$\vec{\mu} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

$\vec{\mu}' = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

Assignment for $X_1 = 0$

Assignment for $X_2 = 1$

Assignment for $X_3 = 0$

$x_1 = 1$

$x_2 = 1$

$x_3 = 0$
Relaxation

$$\ln Z(\theta) = \max_{\mu \in M} \sum_{c \in C} \sum_{x_c} \theta_c(x_c) \mu_c(x_c) + H(\mu)$$

- We still haven’t achieved anything, because:
  1. The marginal polytope $M$ is complex to describe (in general, exponentially many vertices and facets)
  2. $H(\mu)$ is very difficult to compute or optimize over

- We now make two approximations:
  1. We replace $M$ with a relaxation of the marginal polytope, e.g. the local consistency constraints $M_L$
  2. We replace $H(\mu)$ with a function $\tilde{H}(\mu)$ which approximates $H(\mu)$
Local consistency constraints

- Force every “cluster” of variables to choose a local assignment:
  \[
  \mu_i(x_i) \geq 0 \quad \forall i \in V, x_i
  \]
  \[
  \sum_{x_i} \mu_i(x_i) = 1 \quad \forall i \in V
  \]
  \[
  \mu_{ij}(x_i, x_j) \geq 0 \quad \forall ij \in E, x_i, x_j
  \]
  \[
  \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) = 1 \quad \forall ij \in E
  \]

- Enforce that these local assignments are globally consistent:
  \[
  \mu_i(x_i) = \sum_{x_j} \mu_{ij}(x_i, x_j) \quad \forall ij \in E, x_i
  \]
  \[
  \mu_j(x_j) = \sum_{x_i} \mu_{ij}(x_i, x_j) \quad \forall ij \in E, x_j
  \]

- The local consistency polytope, \( M_L \) is defined by these constraints

- **Theorem:** The local consistency constraints exactly define the marginal polytope for a tree-structured MRF
Entropy for tree-structured models

- Suppose that $p$ is a tree-structured distribution, so that we are optimizing only over marginals $\mu_{ij}(x_i, x_j)$ for $ij \in T$

- The solution to $\arg \max q : \mathbb{E}_q[f(x)] = \mu H(q)$ is a tree-structured MRF (c.f. lecture 10, maximum entropy estimation)

- The entropy of $q$ as a function of its marginals can be shown to be

$$H(\vec{\mu}) = \sum_{i \in V} H(\mu_i) - \sum_{ij \in T} I(\mu_{ij})$$

where

$$H(\mu_i) = -\sum_{x_i} \mu_i(x_i) \log \mu_i(x_i)$$

$$I(\mu_{ij}) = \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \log \frac{\mu_{ij}(x_i, x_j)}{\mu_i(x_i) \mu_j(x_j)}$$

- Can we use this for non-tree structured models?
The Bethe entropy approximation is (for any graph)

\[
H_{\text{bethe}}(\vec{\mu}) = \sum_{i \in V} H(\mu_i) - \sum_{ij \in E} I(\mu_{ij})
\]

This gives the following variational approximation:

\[
\max_{\mu \in \mathcal{M}_L} \sum_{c \in C} \sum_{x_c} \theta_c(x_c) \mu_c(x_c) + H_{\text{bethe}}(\vec{\mu})
\]

For non tree-structured models this is not concave, and is hard to maximize.

Loopy belief propagation, if it converges, finds a saddle point!
Concave relaxation

- Let \( \tilde{H}(\mu) \) be an upper bound on \( H(\mu) \), i.e. \( H(\mu) \leq \tilde{H}(\mu) \).
- As a result, we obtain the following upper bound on the log-partition function:
  \[
  \ln Z(\theta) \leq \max_{\mu \in M_L} \sum_{c \in C} \sum_{x_c} \theta_c(x_c) \mu_c(x_c) + \tilde{H}(\mu)
  \]

- An example of a concave entropy upper bound is the tree-rewighted approximation (Jaakkola, Wainwright, & Wilsky, ’05), given by specifying a distribution over spanning trees of the graph:

  ![Spanning Trees Diagram]

  Letting \( \{\rho_{ij}\} \) denote edge appearance probabilities, we have:

  \[
  H_{TRW}(\mu) = \sum_{i \in V} H(\mu_i) - \sum_{ij \in E} \rho_{ij} l(\mu_{ij})
  \]
Comparison of LBP and TRW

We showed two approximation methods, both making use of the local consistency constraints $M_L$ on the marginal polytope:

1. **Bethe-free energy approximation (for pairwise MRFs):**

$$\max_{\mu \in M_L} \sum_{ij \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j) + \sum_{i \in V} H(\mu_i) - \sum_{ij \in E} I(\mu_{ij})$$

- Not concave. Can use concave-convex procedure to find local optima
- Loopy BP, if it converges, finds a saddle point (often a local maxima)

2. **Tree re-weighted approximation (for pairwise MRFs):**

$$(*) \max_{\mu \in M_L} \sum_{ij \in E} \sum_{x_i, x_j} \mu_{ij}(x_i, x_j) \theta_{ij}(x_i, x_j) + \sum_{i \in V} H(\mu_i) - \sum_{ij \in E} \rho_{ij} I(\mu_{ij})$$

- $\{\rho_{ij}\}$ are edge appearance probabilities (must be consistent with some set of spanning trees)
- This is concave! Find global maximiza using projected gradient ascent
- Provides an upper bound on log-partition function, i.e. $\ln Z(\theta) \leq (*)$
Two types of variational algorithms: Mean-field and relaxation

\[
\max_{q \in Q} \sum_{c \in C} \sum_{x_c} q(x_c) \theta_c(x_c) + H(q(x)).
\]

- Although this function is concave and thus in theory should be easy to optimize, we need some compact way of representing \( q(x) \)
- *Relaxation* algorithms work directly with *pseudomarginals* which may not be consistent with any joint distribution
- *Mean-field* algorithms assume a factored representation of the joint distribution, e.g.

\[
q(x) = \prod_{i \in V} q_i(x_i) \quad \text{(called *naive* mean field)}
\]
Naive mean-field

- Using the same notation as in the rest of the lecture, naive mean-field is:

\[
(*) \max_{\mu} \sum_{c \in C} \sum_{x_c} \theta_c(x_c) \mu_c(x_c) + \sum_{i \in V} H(\mu_i) \quad \text{subject to}
\]

\[
\mu_i(x_i) \geq 0 \quad \forall i \in V, x_i \in \text{Val}(X_i)
\]

\[
\sum_{x_i \in \text{Val}(X_i)} \mu_i(x_i) = 1 \quad \forall i \in V
\]

\[
\mu_c(x_c) = \prod_{i \in c} \mu_i(x_i)
\]

- Corresponds to optimizing over an inner bound on the marginal polytope:

- We obtain a lower bound on the partition function, i.e. \((*) \leq \ln Z(\theta)\)