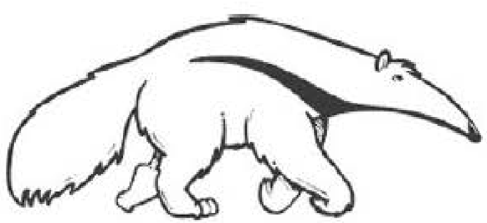
Variational Inference in Probabilistic Graphical Models

Andrew Gelfand

Thursday May 23rd, 2013



BREN:ICS
INFORMATION AND COMPUTER SCIENCES
UNIVERSITY of CALIFORNIA IRVINE

Introduction

- Inference presented algorithmically thus far
  1. Find elimination order
  2. Construct Bucket Tree
  3. Pass messages on Bucket Tree

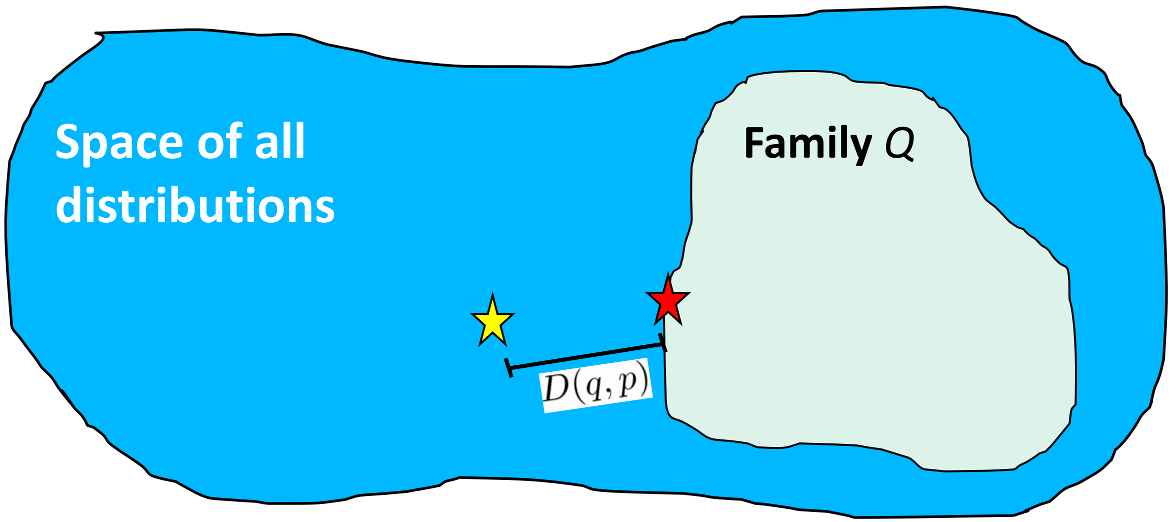
- New perspective on approximation
  - $p(x)$ is hard, so choose an easy $q(x) \in Q$
  - Formulate inference as an optimization problem
    - e.g. minimize "distance" between q and p

Projection Problem

- Given *p*, find distribution from family of distributions *Q* that is closest to *p*:

$$\arg\min_{q \in \mathcal{Q}} D(q, p)$$

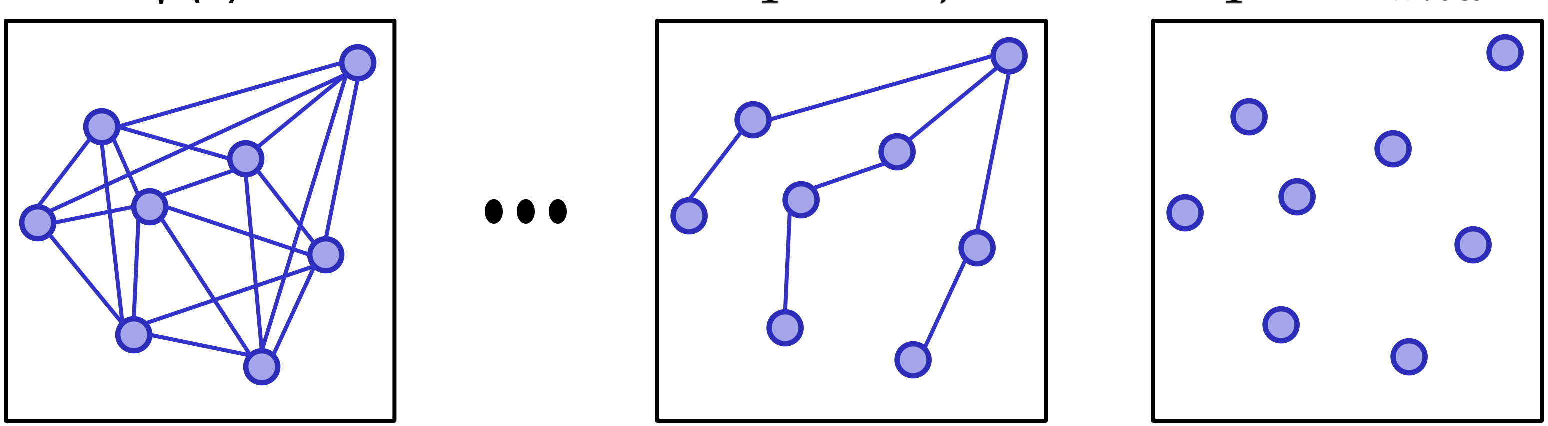
What if $p \in Q$?



Projection Problem

- Given *p*, find distribution from family of distributions *Q* that is closest to *p*

$$\arg\min_{q \in \mathcal{Q}} D(q,p)$$



Outline

- **KL Divergence & Free Energy**
- Simple form of *Q*
  - Mean-Field
  - Exact Inference / Junction Tree
- Approximate Free Energy
  - Loopy Belief Propagation
- Variational Upper Bounds
  - Weighted Mini-Bucket

Divergence Measures

- Say I have distribution $p(x_1, x_2)$:

	x_2	$\bar{x}_2$
x_1	a	b
$\bar{x}_1$	c	d

- Approximate by $q(x_1, x_2) = q(x_1)q(x_2)$

Information-Projection

$$
\begin{aligned}
q^{\star}_{Iproj} &= \arg\min D_{KL}(q, p) \\
&= \arg\min \textstyle\sum_x q(x) \log \left[\frac{q(x)}{p(x)}\right] \\
&= \arg\min -H[q] - E_q\left[\log p\right]
\end{aligned}
$$

$$s.t.\;\; q(x) \geq 0,\; \textstyle\sum_x q(x) = 1$$

Moment-Projection

$$
\begin{aligned}
q^{\star}_{Mproj} &= \arg\min D_{KL}(p, q) \\
&= \arg\min \textstyle\sum_x p(x) \log \left[\frac{p(x)}{q(x)}\right] \\
&= \arg\min -H[p] - E_p\left[\log q\right]
\end{aligned}
$$

$$s.t.\;\; q(x) \geq 0,\; \textstyle\sum_x q(x) = 1$$

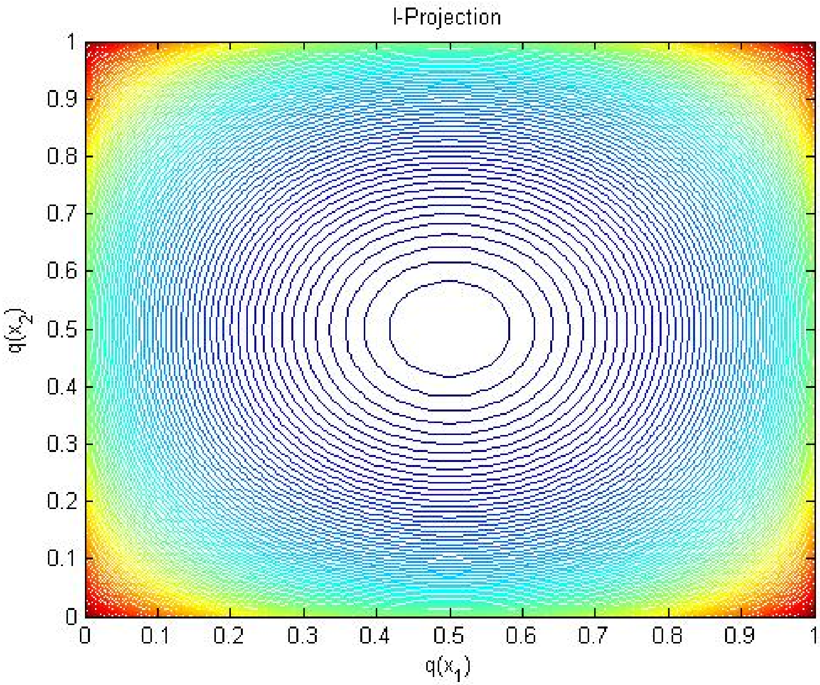
Divergence Measures

- Say I have distribution $p(x_1, x_2)$:

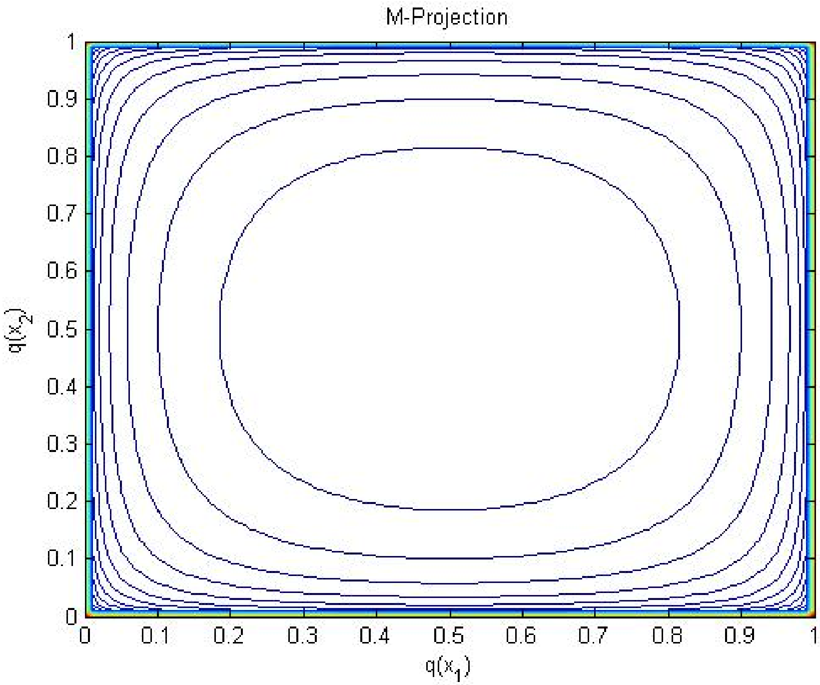
	x_2	$\bar{x}_2$
x_1	0.25	0.25
$\bar{x}_1$	0.25	0.25

- Approximate by $q(x_1, x_2) = q(x_1)q(x_2)$

I-Projection



M-Projection



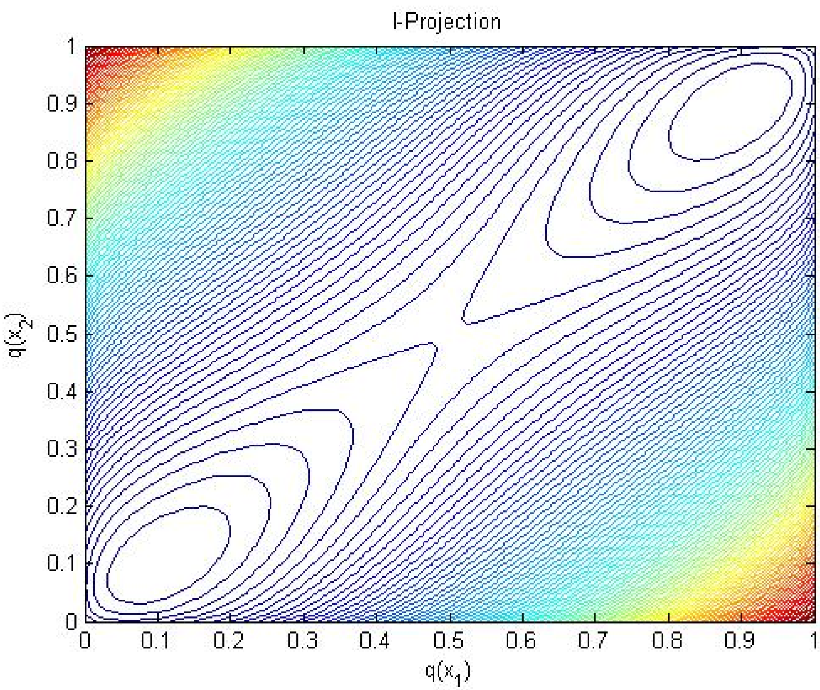
Divergence Measures

- Say I have distribution $p(x_1, x_2)$:

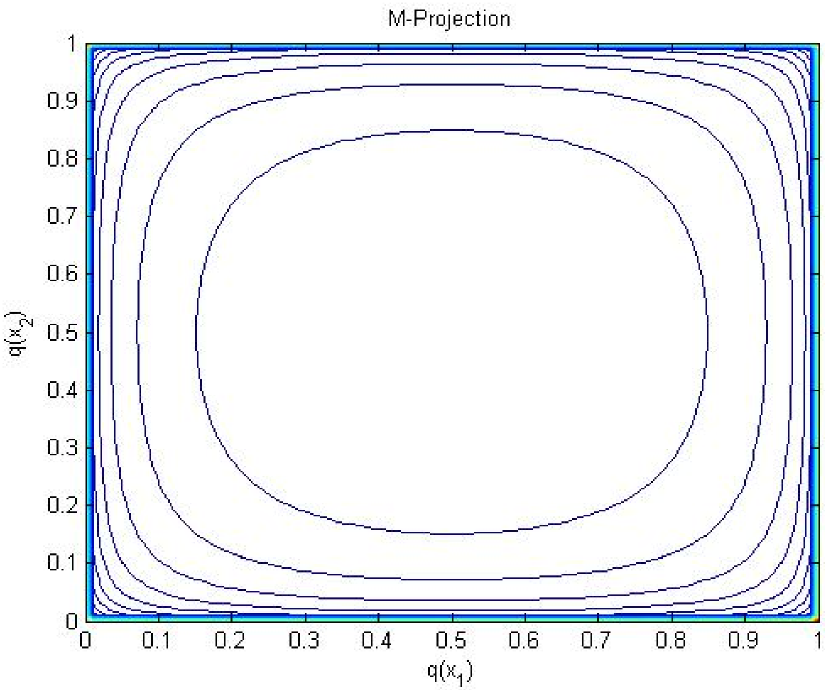
	x_2	$\bar{x}_2$
x_1	0.47	0.03
$\bar{x}_1$	0.03	0.47

- Approximate by $q(x_1, x_2) = q(x_1)q(x_2)$

I-Projection



M-Projection



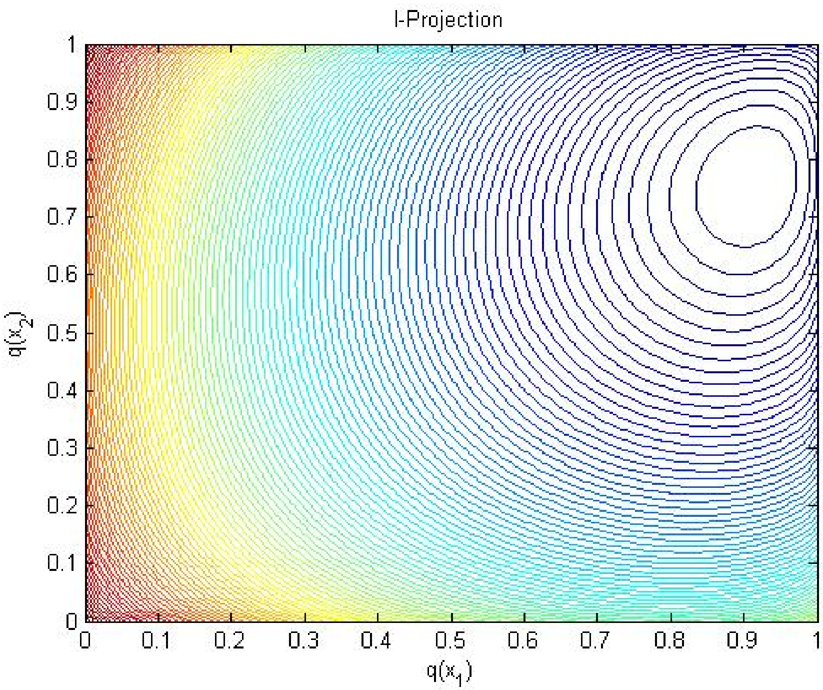
Divergence Measures

- Say I have distribution $p(x_1, x_2)$:

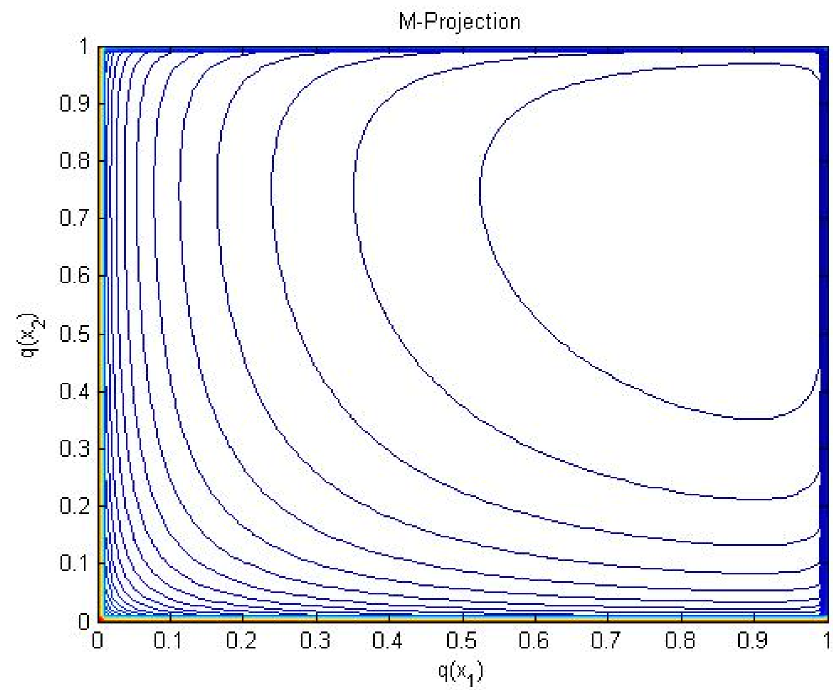
	x_2	$\bar{x}_2$
x_1	0.7	0.2
$\bar{x}_1$	0.05	0.05

- Approximate by $q(x_1, x_2) = q(x_1)q(x_2)$

I-Projection



M-Projection



Free Energy and D_{KL}(q,p)

- Let $p(x) = \frac{1}{Z_\psi} \prod_\alpha \psi_\alpha(x_\alpha) = \frac{1}{Z_\psi} \tilde{p}(x)$ ← Unnormalized Measure
- Consider the I-Projection:

$$D_{KL}(q,p) = \sum_x q(x) \log \left[\frac{q(x)}{p(x)} \right] = -H[q(x)] - \sum_x q(x) \log p(x)$$

$$= -H[q(x)] - \sum_x q(x) \log \tilde{p}(x) + \log Z_\psi$$

- Since $D_{KL}(q,p) \geq 0$ we have a bound

$$\log Z_\psi \geq \underbrace{H[q(x)] + \sum_x q(x) \log \tilde{p}(x)}_{\text{Energy Functional}} := F[q,p]$$

Function is a mapping: $x \mapsto f(x)$
Functional is a mapping: $f \mapsto f(x)$ "function of a function"

Free Energy and $D_{KL}(q,p)$

- Let $p(x) = \frac{1}{Z_\psi} \prod_\alpha \psi_\alpha(x_\alpha) = \frac{1}{Z_\psi} \tilde{p}(x)$ ← Unnormalized Measure
- Consider the I-Projection:

$$D_{KL}(q,p) = \sum_x q(x) \log \left[\frac{q(x)}{p(x)} \right] = -H[q(x)] - \sum_x q(x) \log p(x)$$

- For I-Projections we have:

$$\min_{q \in \mathcal{Q}} D_{KL}(q,p) \equiv \max_{q \in \mathcal{Q}} H[q(x)] + \sum_x q(x) \log \tilde{p}(x)$$

- What about M-Projections?
  - Much harder: requires marginals of $p(x)$

Function is a mapping: $x \mapsto f(x)$
Functional is a mapping: $f \mapsto f(x)$ "function of a function"

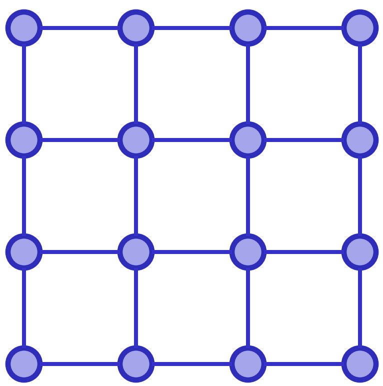
Outline

- KL Divergence & Free Energy
- Simple form of *Q*
  - **Mean-Field**
  - Exact Inference / Junction Tree
- Approximate Free Energy
  - Loopy Belief Propagation
- Variational Upper Bounds
  - Weighted Mini-Bucket

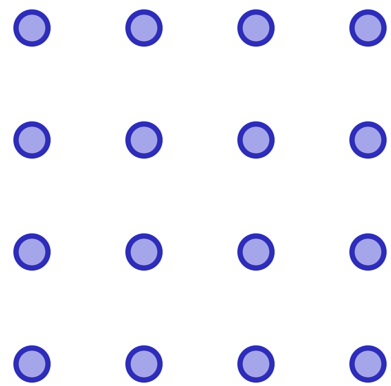
Naïve Mean Field

- Target distribution is: $p(x) = \frac{1}{Z_\psi} \prod_\alpha \psi_\alpha(x_\alpha)$
- Assume *q* takes simple form: $q(x) = \prod_i q_i(x_i)$
- Running Example:

$$p(x) = \frac{1}{Z_\psi} \prod_{ij} \psi_{ij}(x_i, x_j)$$



$$q(x) = \prod_i q_i(x_i)$$



Naïve Mean Field

- Goal is to optimize:

$$\max_{q \in \mathcal{Q}} H[q(x)] + \sum_x q(x) \log \tilde{p}(x)$$

$$\textit{s.t.} \quad q(x) = \prod_i q_i(x_i) \ , \ \sum_{x_i} q_i(x_i) = 1$$

- Can re-write Entropy as:
  - $H[q(x)] = \sum_i H[q_i(x_i)] = -\sum_i \sum_{x_i} q_i(x_i) \log q_i(x_i)$
- For our example: $\tilde{p}(x) = \prod_{ij} \psi_{ij}(x_i, x_j)$
  - $\sum_x q(x) \log \tilde{p}(x) = \sum_{ij} \sum_{x_i, x_j} q_i(x_i) q_j(x_j) \log \psi_{ij}(x_i, x_j)$

Naïve Mean Field

- Construct Lagrangian

$$\mathcal{L} = \sum_i H[q_i(x_i)] + \sum_{ij} \sum_{x_i, x_j} q_i(x_i) q_j(x_j) \log \psi_{ij}(x_i, x_j) + \sum_i \lambda_i \left(\sum_{x_i} q_i(x_i) - 1 \right)$$

From Normalization Constraint

- Take partials and equate to zero

$$\frac{\partial \mathcal{L}}{\partial q_i(x_i)} = -\log q_i(x_i) - 1 + \sum_{j \in N(i)} \sum_{x_j} q_j(x_j) \log \psi_{ij}(x_i, x_j) + \lambda_i = 0$$

Naïve Mean Field

- Construct Lagrangian

$$\mathcal{L} = \sum_i H[q_i(x_i)] + \sum_{ij} \sum_{x_i, x_j} q_i(x_i) q_j(x_j) \log \psi_{ij}(x_i, x_j) + \sum_i \lambda_i \left(\sum_{x_i} q_i(x_i) - 1 \right)$$

From Normalization Constraint

- Take partials and equate to zero

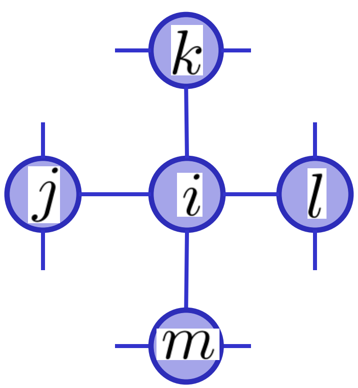
$$\frac{\partial \mathcal{L}}{\partial q_i(x_i)} = -\log q_i(x_i) - 1 + \boxed{\sum_{j \in N(i)} E_q \left[\log \psi_{ij}(x_i, x_j) | X_i = x_i \right]} \lambda_i = 0$$

Naïve Mean Field

- Re-arranging gives:

$$q_i(x_i) = \frac{1}{Z_i} \exp \left\{ \sum_{j \in N(i)} E_q \left[\log \psi_{ij}(x_i, x_j) | X_i = x_i \right] \right\}$$

- For node *i*:



$$q_i(x_i) = \frac{1}{Z_i} \exp \begin{pmatrix} q(x_j) \log \psi_{ij}(x_i, x_j) + q(\bar{x}_j) \log \psi_{ij}(x_i, \bar{x}_j) + \\ q(x_k) \log \psi_{ik}(x_i, x_k) + q(\bar{x}_k) \log \psi_{ik}(x_i, \bar{x}_k) + \\ q(x_l) \log \psi_{il}(x_i, x_l) + q(\bar{x}_l) \log \psi_{il}(x_i, \bar{x}_l) + \\ q(x_m) \log \psi_{im}(x_i, x_m) + q(\bar{x}_m) \log \psi_{im}(x_i, \bar{x}_m) \end{pmatrix}$$

Complexity?

Naïve Mean Field Algorithm

<u>Input</u>: $p(x) = \frac{1}{Z_\psi} \prod_{ij} \psi_{ij}(x_i, x_j)$

<u>Output</u>: $q(x) = \prod_i q_i(x_i)$

initialize each $q_i^{(0)}(x_i)$, $t \leftarrow 0$

Complexity?

while ¬*converged*

  for each node *i*

    Update: $q_i^{(t+1)}(x_i) \leftarrow \exp\left\{\sum_{j \in N(i)} E_q\left[\log \psi_{ij}(x_i, x_j) | X_i = x_i\right]\right\}$

    Normalize: $q_i^{(t+1)}(x_i)$

  $t \leftarrow t + 1$

return $q(x)$

Naïve Mean Field Summary

- Every update increases energy
  - Look at terms involving q_i

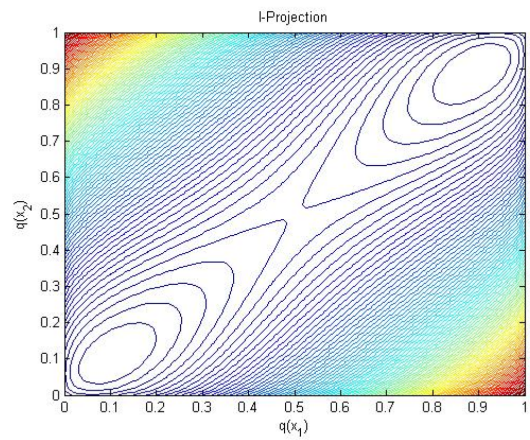
$$F[q_i,p] = \underbrace{H_i[q_i(x_i)]}_{\text{Concave in } q_i(x_i)} + \underbrace{\sum_{j\in N(i)} \sum_{x_i,x_j} q_i(x_i)q_j(x_j)\log\psi_{ij}(x_i,x_j)}_{\text{Linear in } q_i(x_i)}$$

- Will converge to stationary point
- Limitations:

If $p(x_1,x_2)$:

	x_2	$\bar{x}_2$
x_1	0.47	0.03
$\bar{x}_1$	0.03	0.47

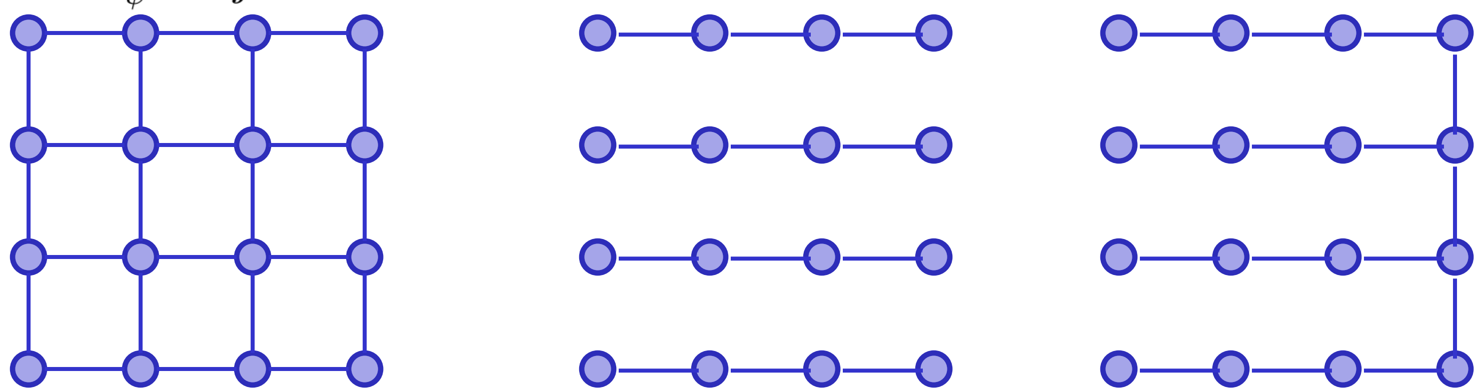
then



Structured Mean Field

- Choose *q* with some low tree-width structure
  - Updates more complex / require inference in *q*

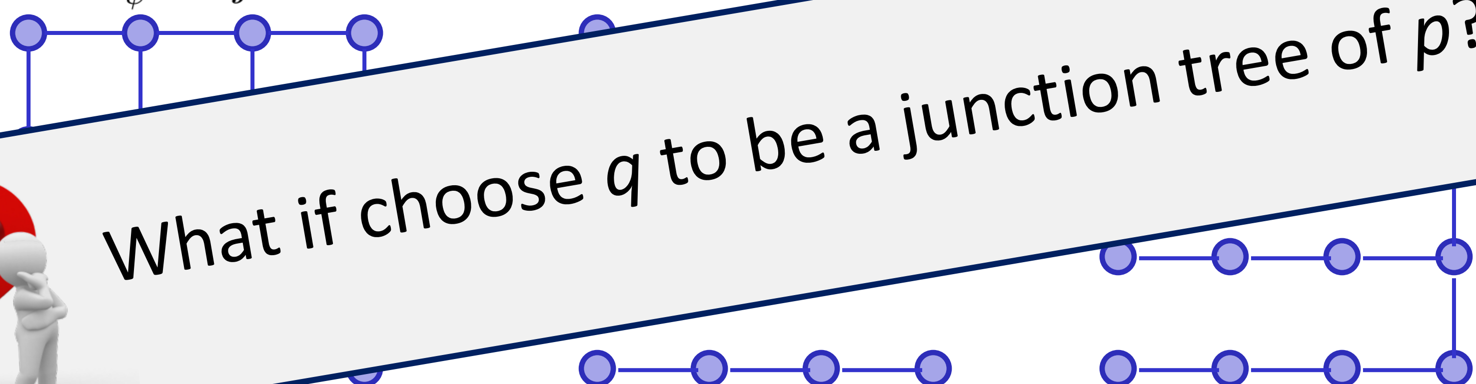
$p(x) = \frac{1}{Z_\psi} \prod_{ij} \psi_{ij}(x_i, x_j)$ $\quad q'(x)$ $\quad q^\dagger(x)$



Structured Mean Field

- Choose *q* with some low tree-width structure
  - Updates more complex / require inference in *q*

$p(x) = \frac{1}{Z_\psi} \prod_{ij} \psi_{ij}(x_i, x_j)$ $\qquad$ $q'(x)$



What if choose *q* to be a junction tree of *p*?

Outline

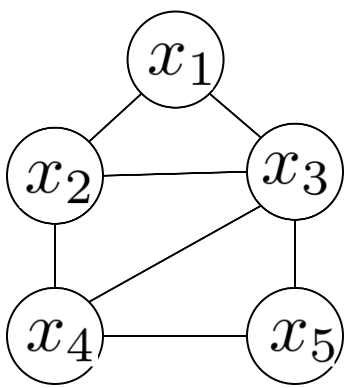
- KL Divergence & Free Energy
- Simple form of *Q*
  - Mean-Field
  - **Exact Inference / Junction Tree**
- Approximate Free Energy
  - Loopy Belief Propagation
- Variational Upper Bounds
  - Weighted Mini-Bucket

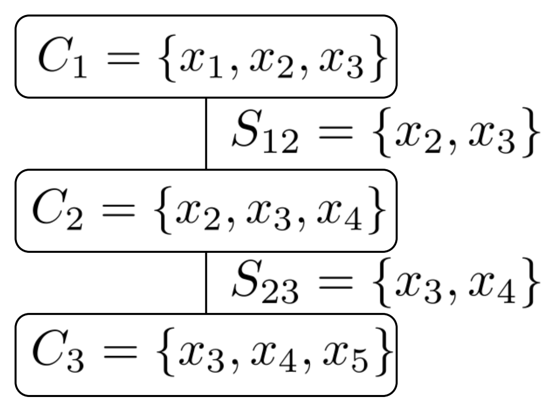
Junction Trees

Recall that:

Junction Trees Satisfy:
- Factor Preservation
- Running Intersection

- Let T be a junction tree of $p(x)$
  - Let C_i denote clusters in T
  - Let S_{ij} denote separators on edges of T
  - Let β_i be the belief over cluster C_i
  - Let μ_{ij} be the belief over edge sep. S_{ij}
- Ex: $p(x_1, x_2, x_3, x_4, x_5) = \frac{1}{Z_\psi}\psi_{12}(x_1, x_2)\psi_{13}(x_1, x_3)\cdots\psi_{45}(x_4, x_5)$





Junction Trees

- Junction Tree *T* of *p*(*x*) defines a distribution $q_T(x)$

$$q_{\mathcal{T}}(x) = \frac{\prod_{i \in V_{\mathcal{T}}} \beta_i(x_{c_i})}{\prod_{ij \in E_{\mathcal{T}}} \mu_{ij}(x_{s_{ij}})}$$

  where

$$\sum_{x_{c_i} \setminus x_{s_{ij}}} \beta_i(x_{c_i}) = \mu_{ij}(x_{s_{ij}}) = \sum_{x_{c_j} \setminus x_{s_{ij}}} \beta_j(x_{c_j})$$

- 'Consistent' beliefs are marginals of $q_T(x)$
  - e.g., $q_{\mathcal{T}}(x_1, x_2, x_3) = \beta_1(x_1, x_2, x_3)$

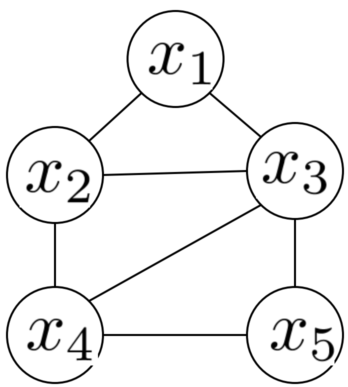
Junction Trees

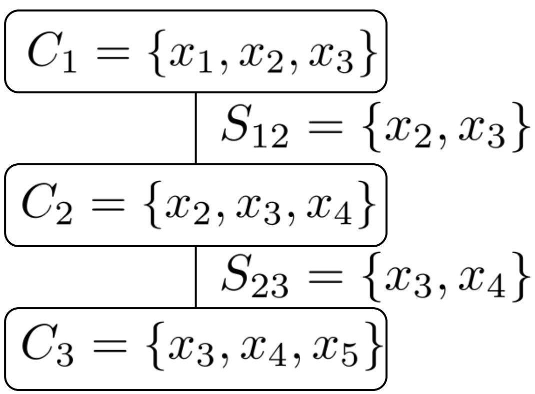
- In our example:

$$q_{\mathcal{T}}(x_1, x_2, x_3, x_4, x_5) = \frac{\beta_1(x_1,x_2,x_3)\beta_2(x_2,x_3,x_4)\beta_3(x_3,x_4,x_5)}{\mu_{12}(x_2,x_3)\mu_{23}(x_3,x_4)}$$

where $\sum_{x_1} \beta_1(x_1, x_2, x_3) = \mu_{12}(x_2, x_3) = \sum_{x_4} \beta_2(x_2, x_3, x_4)$

$\sum_{x_2} \beta_2(x_2, x_3, x_4) = \mu_{23}(x_3, x_4) = \sum_{x_5} \beta_3(x_3, x_4, x_5)$





Exact Inference as Optimization

- Goal is to optimize:

$$\max_{q_{\mathcal{T}}} F\left[(q_{\mathcal{T}}(x_1, ..., x_5), p(x_1, ..., x_5)\right]$$

s.t. $q_{\mathcal{T}}(x_1, x_2, x_3, x_4, x_5) = \frac{\beta_1(x_1,x_2,x_3)\beta_2(x_2,x_3,x_4)\beta_3(x_3,x_4,x_5)}{\mu_{12}(x_2,x_3)\mu_{23}(x_3,x_4)}$

$$\sum_{x_1} \beta_1(x_1, x_2, x_3) = \mu_{12}(x_2, x_3) = \sum_{x_4} \beta_2(x_2, x_3, x_4)$$

$$\sum_{x_2} \beta_2(x_2, x_3, x_4) = \mu_{23}(x_3, x_4) = \sum_{x_5} \beta_3(x_3, x_4, x_5)$$

Consistency Constraints

$$\sum_{x_1,x_2,x_3} \beta_1(x_1, x_2, x_3) = 1$$

$$\sum_{x_2,x_3,x_4} \beta_2(x_2, x_3, x_4) = 1$$

$$\sum_{x_3,x_4,x_5} \beta_3(x_3, x_4, x_5) = 1$$

Normalization Constraints

$$\beta_1(x_1, x_2, x_3) \geq 0, \beta_2(x_2, x_3, x_4) \geq 0, \beta_3(x_3, x_4, x_5) \geq 0$$

Exact Inference as Optimization

- More generally:

$$\max_{q \in \mathcal{Q}_\mathcal{T}} H[q(x)] + \sum_x q(x) \log \tilde{p}(x)$$

s.t.

$$\sum_{x_{c_i} \setminus x_{s_{ij}}} \beta_i(x_{c_i}) = \mu_{ij}(x_{s_{ij}}) \quad \text{for all edges}$$

$$\sum_{x_{c_i}} \beta_i(x_{c_i}) = 1$$

$$\beta_i(x_{c_i}) \geq 0$$

for all vertices

- Find stationary points by:
  - Constructing Lagrangian $\mathcal{L}$
  - Taking derivatives of $\mathcal{L}$ wrt $\beta_i(x_{c_i}), \mu_{ij}(x_{s_{ij}})$

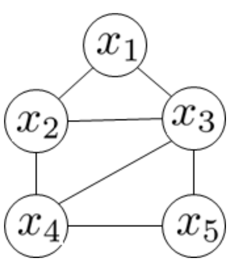
Fixed Point Characterization

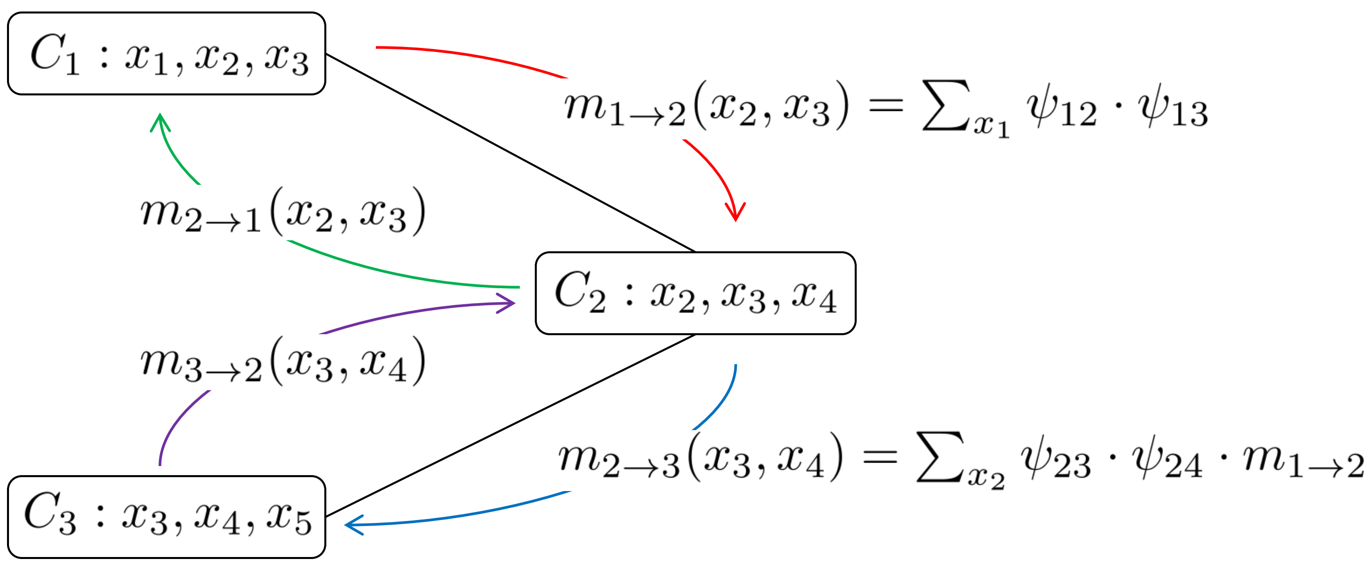
- Results in standard message passing updates:

$$m_{i \to j}(x_{s_{ij}}) \propto \sum_{x_{c_i} \setminus x_{s_{ij}}} \psi_i(x_{c_i}) \prod_{k \in N(i) \setminus j} m_{k \to i}(x_{s_{ik}})$$

$$b_i(x_{c_i}) \propto \psi_i(x_{c_i}) \prod_{j \in N(i)} m_{j \to i}(x_{s_{ij}})$$

- Ex:





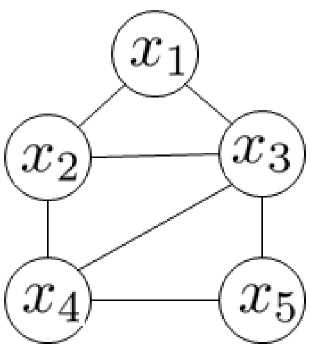
Outline

- KL Divergence & Free Energy
- Simple form of *Q*
  - Mean-Field
  - Exact Inference / Junction Tree
- Approximate Free Energy
  - **Loopy Belief Propagation**
- Variational Upper Bounds
  - Weighted Mini-Bucket

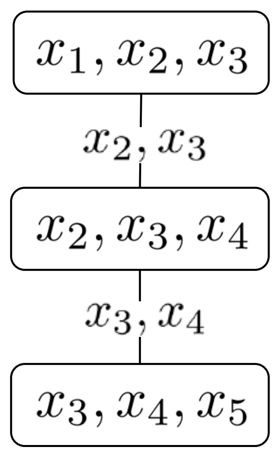
Loopy Belief Propagation

- Cluster Graph generalizes Junction Tree
  - Family preservation & *relaxed* running intersection

Markov Network

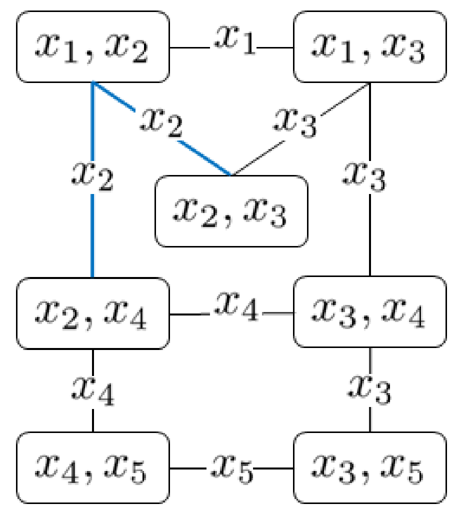


Junction Tree



$$\boxed{S_{ij} = C_i \cap C_j}$$

Loopy Cluster Graph



$$\boxed{S_{ij} \subseteq C_i \cap C_j}$$

Factored Energy Functional

- Exact inference was cast as:

$$\max_q F[q,p] = \max_q H[q(x)] + \sum_x q(x) \log \tilde{p}(x)$$

s.t.

$$q_{\mathcal{T}}(x) = \frac{\prod_{i \in V_{\mathcal{T}}} \beta_i(x_{c_i})}{\prod_{ij \in E_{\mathcal{T}}} \mu_{ij}(x_{s_{ij}})}$$

$$\sum_{x_{c_i} \setminus x_{s_{ij}}} \beta_i(x_{c_i}) = \mu_{ij}(x_{s_{ij}}) \quad \text{for all edges}$$

$$\sum_{x_{c_i}} \beta_i(x_{c_i}) = 1 \quad \text{for all vertices}$$

$$\beta_i(x_{c_i}) \geq 0$$

Factored Energy Functional

- Exact inference was cast as:

$$\max_q F[q,p] = \max_q H[q(x)] + \sum_x q(x) \log \tilde{p}(x)$$

- Because *q* is a junction tree, entropy decomposes

$$H[q(x)] = \sum_{i \in V_\mathcal{T}} H[\beta_i] - \sum_{ij \in E_\mathcal{T}} H[\mu_{ij}]$$

Why?

- Factored Energy

$$\tilde{F}[q,p] = \sum_{i \in V_\mathcal{T}} H[\beta_i] - \sum_{ij \in E_\mathcal{T}} H[\mu_{ij}] + \mathbf{E}_q[\log \tilde{p}]$$

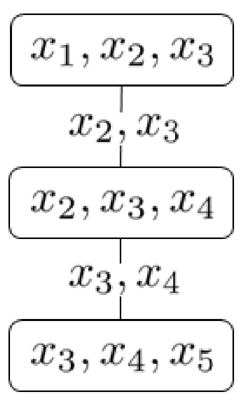
What if *q* isn't a junction tree?

$$\max_q \tilde{F}[q,p]$$

s.t. $\sum_{x_{c_i} \setminus x_{s_{ij}}} \beta_i(x_{c_i}) = \mu_{ij}(x_{s_{ij}})$, $\sum_{x_{c_i}} \beta_i(x_{c_i}) = 1$

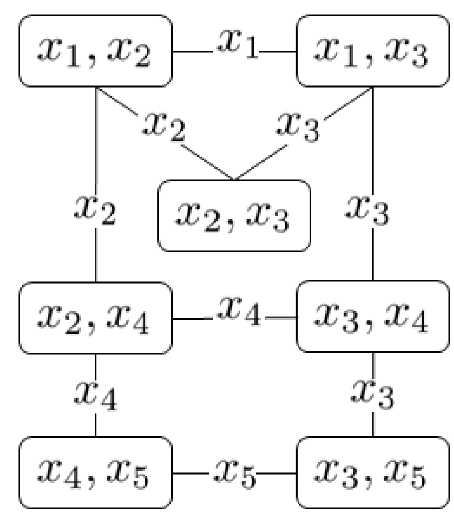
$$q_{\mathcal{T}}(x) = \frac{\prod_{i \in V_{\mathcal{T}}} \beta_i(x_{c_i})}{\prod_{ij \in E_{\mathcal{T}}} \mu_{ij}(x_{s_{ij}})}$$

Junction Tree



$$q(x) = \frac{\prod_{i \in V} \beta_i(x_{c_i})}{\prod_{ij \in E} \mu_{ij}(x_{s_{ij}})}$$

Loopy Cluster Graph



What if *q* isn't a junction tree?

$$\max_q \tilde{F}[q,p]$$

s.t. $\sum_{x_{c_i} \setminus x_{s_{ij}}} \beta_i(x_{c_i}) = \mu_{ij}(x_{s_{ij}})$, $\sum_{x_{c_i}} \beta_i(x_{c_i}) = 1$

$$q_{\mathcal{T}}(x) = \frac{\prod_{i \in V_{\mathcal{T}}} \beta_i(x_{c_i})}{\prod_{ij \in E_{\mathcal{T}}} \mu_{ij}(x_{s_{ij}})}$$

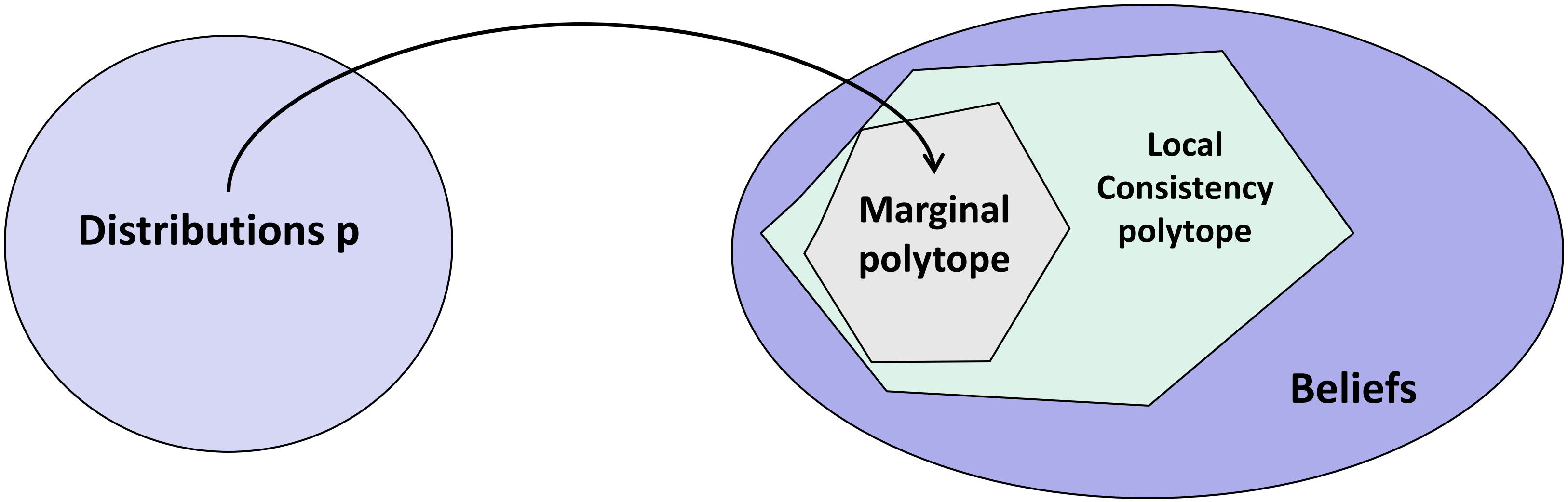
- Beliefs are marginals
  - $q(x_{c_i}) = \beta_i(x_{c_i})$
- Entropy factors, so $\tilde{F}[q,p]$ is exact energy
- Bound on log Z_ψ

$$q(x) = \frac{\prod_{i \in V} \beta_i(x_{c_i})}{\prod_{ij \in E} \mu_{ij}(x_{s_{ij}})}$$

- Beliefs not necessarily marginals
- Entropy doesn't factor, so $\tilde{F}[q,p] \approx F[q,p]$
- **No** bound on log Z_ψ

Marginal Polytope

- Marginal Polytope: Set of *achievable* marginals
  - Not compact generally (exponential # of constraints)
  - Difficult to optimize over
  - NP-hard even to check if beliefs lie in polytope



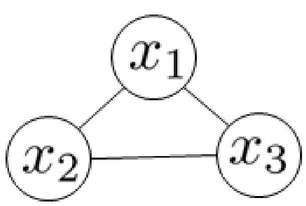
Cartoon borrowed from Andrew McCallum
(http://people.cs.umass.edu/~mccallum/courses/gm2011/14-loopy-bp.pdf)

Marginal vs. Local Polytope

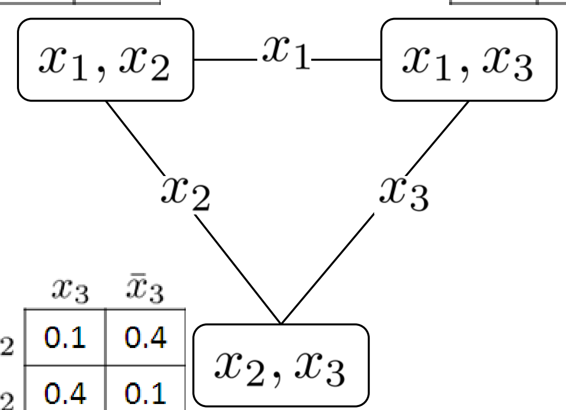
- Local consistency polytope defined by
  - $\sum_{x_{c_i} \setminus x_{s_{ij}}} \beta_i(x_{c_i}) = \mu_{ij}(x_{s_{ij}})$, $\sum_{x_{c_i}} \beta_i(x_{c_i}) = 1$, $\beta_i(x_{c_i}) \geq 0$

- Example:

Markov Network



Locally Consistent set of Beliefs



β_{12} :

	x_2	$\bar{x}_2$
x_1	0.4	0.1
$\bar{x}_1$	0.1	0.4

β_{13} :

	x_3	$\bar{x}_3$
x_1	0.4	0.1
$\bar{x}_1$	0.1	0.4

β_{23} :

	x_3	$\bar{x}_3$
x_2	0.1	0.4
$\bar{x}_2$	0.4	0.1

Marginal vs. Local Polytope

- Local consistency polytope defined by
  - $\sum_{x_{c_i} \backslash x_{s_{ij}}} \beta_i(x_{c_i}) = \mu_{ij}(x_{s_{ij}})$, $\sum_{x_{c_i}} \beta_i(x_{c_i}) = 1$, $\beta_i(x_{c_i}) \geq 0$

Try to find a solution to:

1: $p(x_1, x_2, x_3) + p(x_1, x_2, \bar{x}_3) = 0.4$

2: $p(x_1, x_2, x_3) + p(x_1, \bar{x}_2, x_3) = 0.4$

3: $p(x_1, x_2, x_3) + p(\bar{x}_1, x_2, x_3) = 0.1$

4: $p(x_1, \bar{x}_2, \bar{x}_3) + p(x_1, x_2, \bar{x}_3) = 0.1$

5: $p(x_1, \bar{x}_2, x_3) + p(x_1, \bar{x}_2, \bar{x}_3) = 0.1$

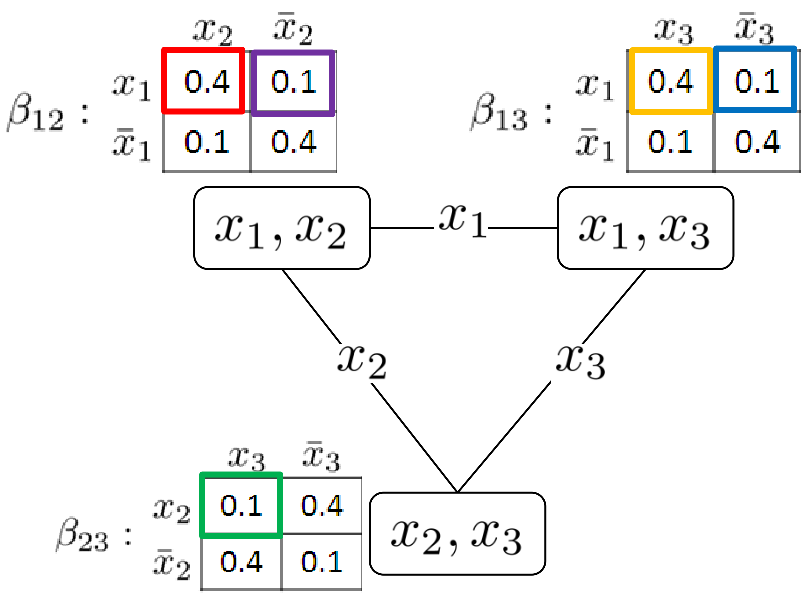
$$p(x_1, x_2, x_3) \geq 0$$

$$\vdots$$

$$p(\bar{x}_1, \bar{x}_2, \bar{x}_3) \geq 0$$

$$\sum_{x_1, x_2, x_3} p(x_1, x_2, x_3) = 1$$

Locally Consistent set of Beliefs



Marginal vs. Local Polytope

- Local consistency polytope defined by
  - $\sum_{x_{c_i} \setminus x_{s_{ij}}} \beta_i(x_{c_i}) = \mu_{ij}(x_{s_{ij}})$, $\sum_{x_{c_i}} \beta_i(x_{c_i}) = 1$, $\beta_i(x_{c_i}) \geq 0$

Try to find a solution to:

$p(x_1, x_2, x_3) + p(x_1, x_2, \bar{x}_3) = 0.4$

$p(x_1, x_2 \ldots$

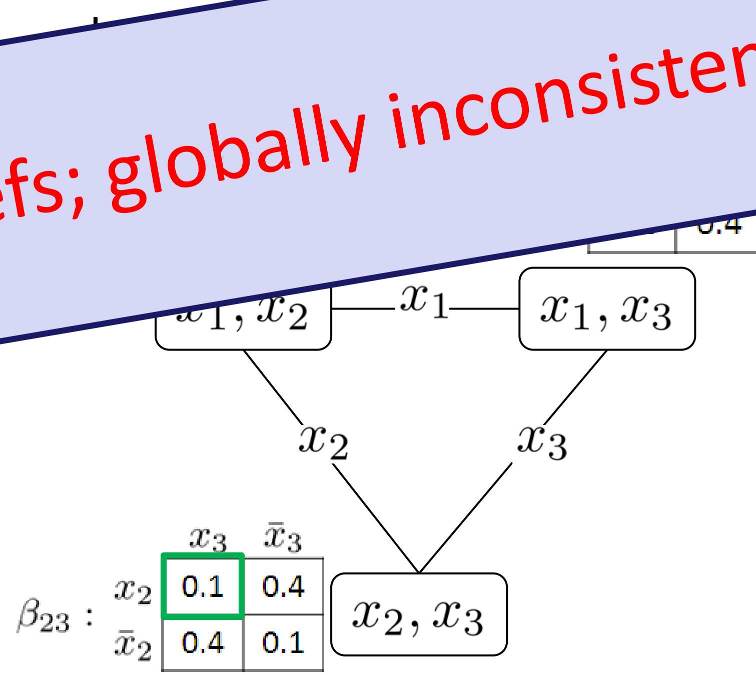
$\ldots = 0.1$

$p(x_1, x_2, x_3) \geq 0$

$\vdots$

$p(\bar{x}_1, \bar{x}_2, \bar{x}_3) \geq 0$

$\sum_{x_1, x_2, x_3} p(x_1, x_2, x_3) = 1$



Locally consistent beliefs; globally inconsistent !!

Loopy BP Algorithm

Input: $p(x) = \frac{1}{Z_\psi} \prod_{ij} \psi_{ij}(x_i, x_j)$

Output: Approximate marginals $\beta_i(x_{c_i})$

build cluster graph: $CG = (V, E)$

initialize messages: $m_{i \to j}(x_{s_{ij}}) = \mathbf{1}$

while locally inconsistent beliefs

- **for** each edge $(i \to j) \in E$
  - update message: $m_{i \to j}(x_{s_{ij}}) \propto \sum_{x_{c_i} \setminus x_{s_{ij}}} \psi_i(x_{c_i}) \prod_{k \in N(i) \setminus j} m_{k \to i}(x_{s_{ik}})$
- **for** each node
  - update beliefs: $b_i(x_{c_i}) \propto \psi_i(x_{c_i}) \prod_{j \in N(i)} m_{j \to i}(x_{s_{ij}})$

Loopy BP Summary

- Introduces **two** approximations
  - Inexact, factored energy functional
  - Local consistency may yield *bad* marginals
- Does not provide a bound on log Z_ψ
- Does not improve energy at every iteration
- Might not converge, many stationary points
- Useful in *hard* problems!
  - Easy to implement / solid empirical performance

Outline

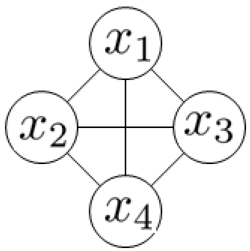
- KL Divergence & Free Energy
- Simple form of *Q*
  - Mean-Field
  - Exact Inference / Junction Tree
- Approximate Free Energy
  - Loopy Belief Propagation
- Variational Upper Bounds
  - **Weighted Mini-Bucket**

Weighted Mini-Bucket [Liu & Ihler]

- Builds upon Mini-Bucket Elimination (MBE)
- Bounds log Z_ψ using Hölder's Inequality
  - Parameterized by set of weights
  - Weights optimized to 'tighten' bound
  - Standard MBE is specific setting of weights
- Complexity controlled by *iBound* parameter

Review of Mini-Bucket Elimination

Markov Network



$$Z_\psi = \sum_{x_1,x_2,x_3,x_4} \psi_{12}\psi_{13}\psi_{14}\psi_{23}\psi_{24}\psi_{34}$$

$$= \sum_{x_4} \sum_{x_3} \psi_{34} \sum_{x_2} \psi_{23}\psi_{24} \underbrace{\sum_{x_1} \psi_{12}\psi_{13}\psi_{14}}_{\text{Bucket 1}}$$

In Bucket 1:

Copies of Variable x_1

$$\sum_{x_1} \underbrace{\psi_{12}(x_1,x_2)\psi_{13}(x_1,x_3)\psi_{14}(x_1,x_4)}_{\text{Cost is } O(k^4)} \le \sum_{x_1^1} \underbrace{\psi_{12}(x_1^1,x_2)\psi_{13}(x_1^1,x_3)}_{\text{Cost is } O(k^3)} \max_{x_1^2} \psi_{14}(x_1^2,x_4)$$

Hölder's Inequality

- The weighted summation operator is:

$$\sum_x^{w_i} f_i(x) := \left(\sum_x f_i(x)^{1/w_i} \right)^{w_i}$$

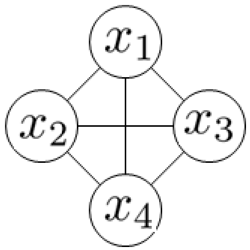
where $f_i(x)$, $i = 1...m$ are positive functions and $w = [w_1, .., w_m]$ are weights

- Hölder's Inequality
  - Let $w_0 = \sum_i w_i$ and all weights be positive, then

$$\sum_x^{w_0} \prod_i f_i(x) \leq \prod_i \sum_x^{w_i} f_i(x) = \prod_i \left(\sum_x f_i(x)^{1/w_i} \right)^{w_i}$$

Weighted Mini-Bucket Elimination

Markov Network



$$Z_\psi = \sum_{x_1,x_2,x_3,x_4} \psi_{12}\psi_{13}\psi_{14}\psi_{23}\psi_{24}\psi_{34}$$

$$= \sum_{x_4} \sum_{x_3} \psi_{34} \sum_{x_2} \psi_{23}\psi_{24} \sum_{x_1} \psi_{12}\psi_{13}\psi_{14}$$

In Bucket 1:

Let $w_1 + w_2 = 1$

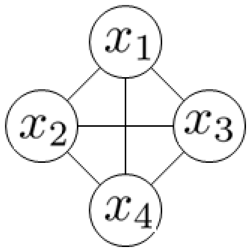
$$\sum_{x_1} \psi_{12}(x_1,x_2)\psi_{13}(x_1,x_3)\psi_{14}(x_1,x_4) \le \sum_{x_1^1}^{w_1} \psi_{12}(x_1^1,x_2)\psi_{13}(x_1^1,x_3) \sum_{x_1^2}^{w_2} \psi_{14}(x_1^2,x_4)$$

Gives the mini-bucket bound:

$$Z_\psi \le \sum_{x_2,x_3,x_4} \psi_{34}\psi_{23}\psi_{24} \sum_{x_1^1}^{w_1} \psi_{12}(x_1^1,x_2)\psi_{13}(x_1^1,x_3) \sum_{x_1^2}^{w_2} \psi_{14}(x_1^2,x_4)$$

Weighted Mini-Bucket Elimination

Markov Network



$$Z_\psi = \sum_{x_1,x_2,x_3,x_4} \psi_{12}\psi_{13}\psi_{14}\psi_{23}\psi_{24}\psi_{34}$$

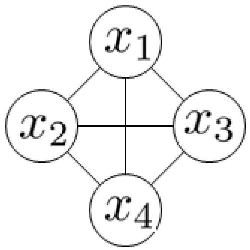
$$= \sum_{x_4} \sum_{x_3} \psi_{34} \sum_{x_2} \psi_{23}\psi_{24} \sum_{x_1} \psi_{12}\psi_{13}\psi_{14}$$

- What happens when?

$$\lim_{w_2 \to 0^+} \sum_{x_1^1}^{w_1} \psi_{12}(x_1^1, x_2)\psi_{13}(x_1^1, x_3) \sum_{x_1^2}^{w_2} \psi_{14}(x_1^2, x_4)$$

Weighted Mini-Bucket Elimination

Markov Network



$$Z_\psi = \sum_{x_1,x_2,x_3,x_4} \psi_{12}\psi_{13}\psi_{14}\psi_{23}\psi_{24}\psi_{34}$$

$$= \sum_{x_4} \sum_{x_3} \psi_{34} \sum_{x_2} \psi_{23}\psi_{24} \sum_{x_1} \psi_{12}\psi_{13}\psi_{14}$$

- What happens when?

$$\lim_{w_2 \to 0^+} \sum_{x_1^1}^{w_1} \psi_{12}(x_1^1, x_2)\psi_{13}(x_1^1, x_3) \sum_{x_1^2}^{w_2} \psi_{14}(x_1^2, x_4)$$

$$=$$

"Standard " mini-Bucket

$$\sum_{x_1^1} \psi_{12}(x_1^1, x_2)\psi_{13}(x_1^1, x_3) \max_{x_1^2} \psi_{14}(x_1^2, x_4)$$

One-Pass WMB Algorithm

<u>Input</u>: $p(x) = \frac{1}{Z_\psi} \prod_\alpha \psi_\alpha(x_\alpha)$, elimination order *o*

<u>Output</u>: Partition function bound $\hat{Z}_\psi(w) \geq Z_\psi$

set $F = \{\psi_\alpha\}$

for *i*=1…*n* along ordering *o*

$$B_i \leftarrow \{\psi_\alpha | \psi_\alpha \in F, x_i \in x_\alpha\}, \;\; F \leftarrow F - B_i$$

Partition B_i into R_i mini-buckets s.t. $B_i = \cup B_{i^r}$

Assign weight w_{i^r} to each B_{i^r} s.t. $\sum_{r=1}^{R_i} w_{i^r} = 1$

$$F \leftarrow F \cup \{\sum_{x_{i^r}}^{w_{i^r}} \prod_{\psi \in B_{i^r}} \psi\}$$

return $\hat{Z}_\psi(w) = \prod_{\psi \in F} \psi$

Tightening the bound

- Note that bound written as $\hat{Z}_\psi(w) \geq Z_\psi$
- Let $\mathcal{D}(w) = \left\{ w | \sum_r w_{i^r} = 1, w_{i^r} \geq 0 \forall i \right\}$
- Optimization problem is:

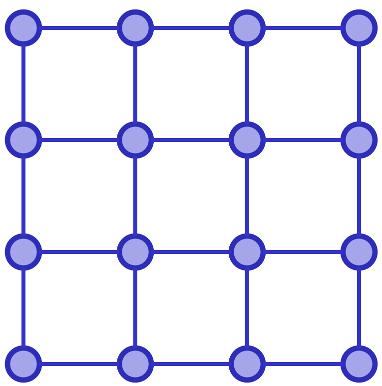
$$\min_w \hat{Z}_\psi(w) \;\; s.t. \; w \in \mathcal{D}(w)$$

- Weights are optimized by iterative algorithm
  - Messages passed up/down the bucket tree
  - Weights updated on each pass

Experiments

- Run on 15-by-15 grid with binary variables

$$p(x) = \frac{1}{Z_\psi} \prod_{i \in V} \psi_i \prod_{ij \in E} \psi_{ij}$$



$$\psi_i(x_i) = \exp(\theta_i(I[x_i = 0] - I[x_i = 1]))$$

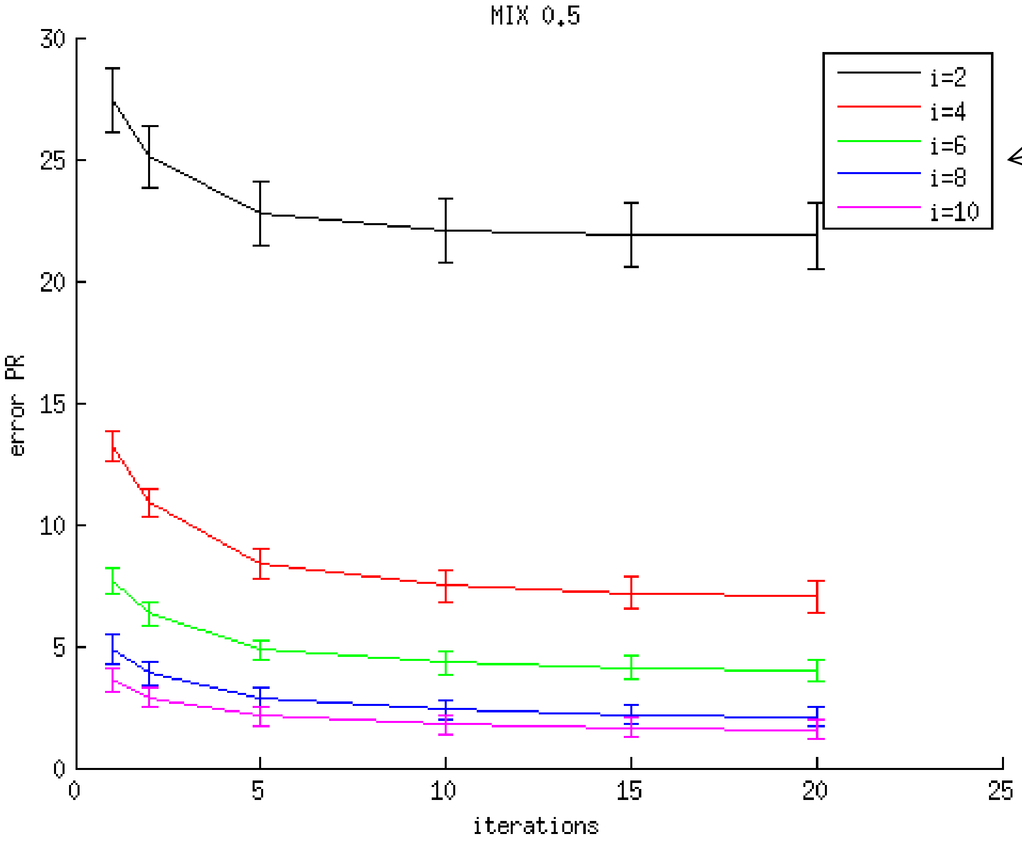
$$\psi_{ij}(x_i, x_j) = \exp\left(\theta_{ij}(I[x_i = x_j] - I[x_i \neq x_j])\right)$$

$\theta_i \sim \mathcal{N}(0, 0.1)$

$\theta_{ij} \sim \mathcal{N}(0, \sigma^2)$ for $\sigma^2 \in \{0.5, 1, 2\}$

WMB - $\theta_{ij} \sim \mathcal{N}(0, 0.5)$

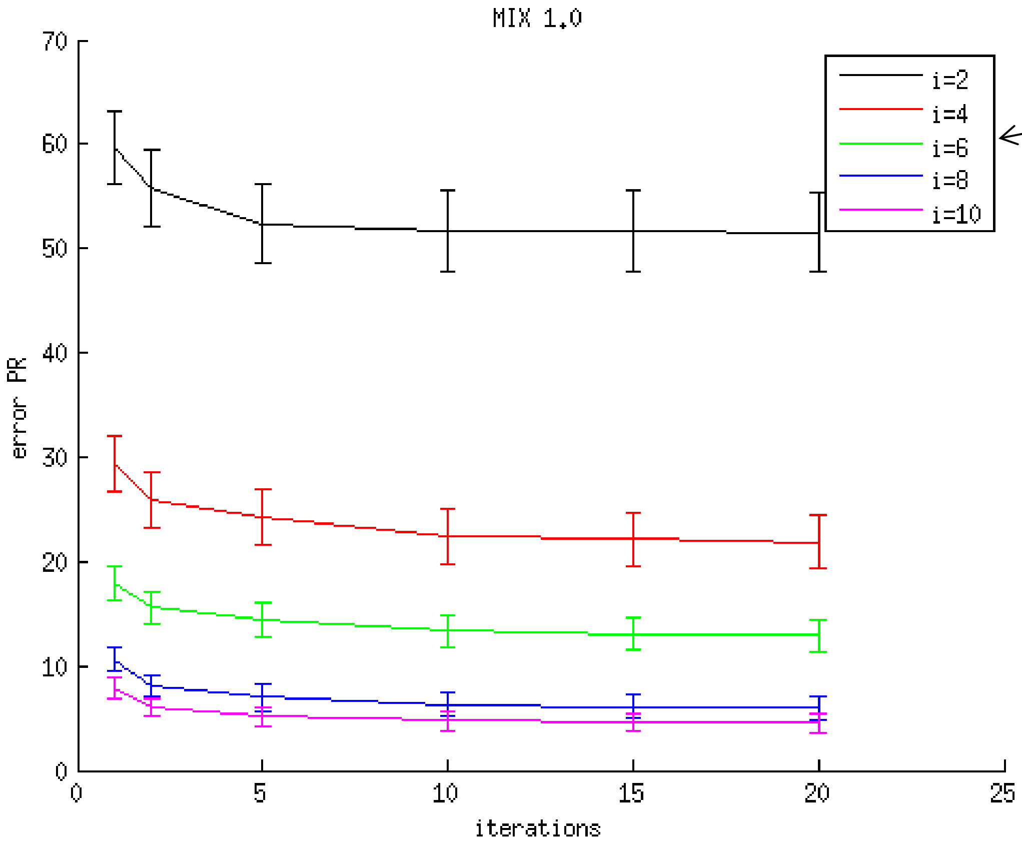
$$\log \left[\frac{\hat{Z}_\psi(w)}{Z_\psi} \right]$$



iBounds

WMB - $\theta_{ij} \sim \mathcal{N}(0,1)$

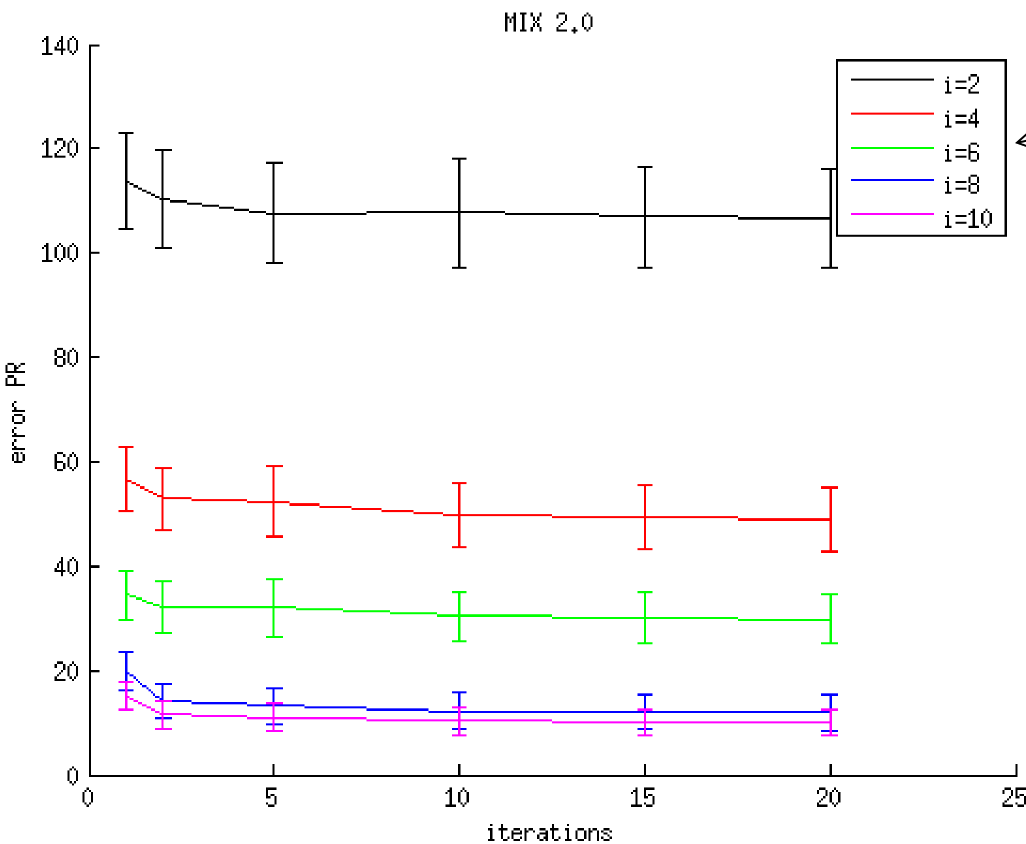
$$\log\left[\frac{\hat{Z}_\psi(w)}{Z_\psi}\right]$$



iBounds

WMB - $\theta_{ij} \sim \mathcal{N}(0,2)$

$$\log\left[\frac{\hat{Z}_\psi(w)}{Z_\psi}\right]$$



iBounds

Summary

- Variational methods formulate inference as an optimization problem
  - e.g. given *p*, find distribution in *Q* closest to *p*
- Provides new perspective for analysis
  - e.g. equivalence between fixed points of sum-product message passing and stationary points
- Led to development of many new algorithms
  - e.g. Liu & Ihler's weighted mini-bucket