Probabilistic & Unsupervised Learning

Convex Algorithms in Approximate Inference

Yee Whye Teh
ywteh@gatsby.ucl.ac.uk

Gatsby Computational Neuroscience Unit, and
MSc ML/CSML, Dept Computer Science
University College London

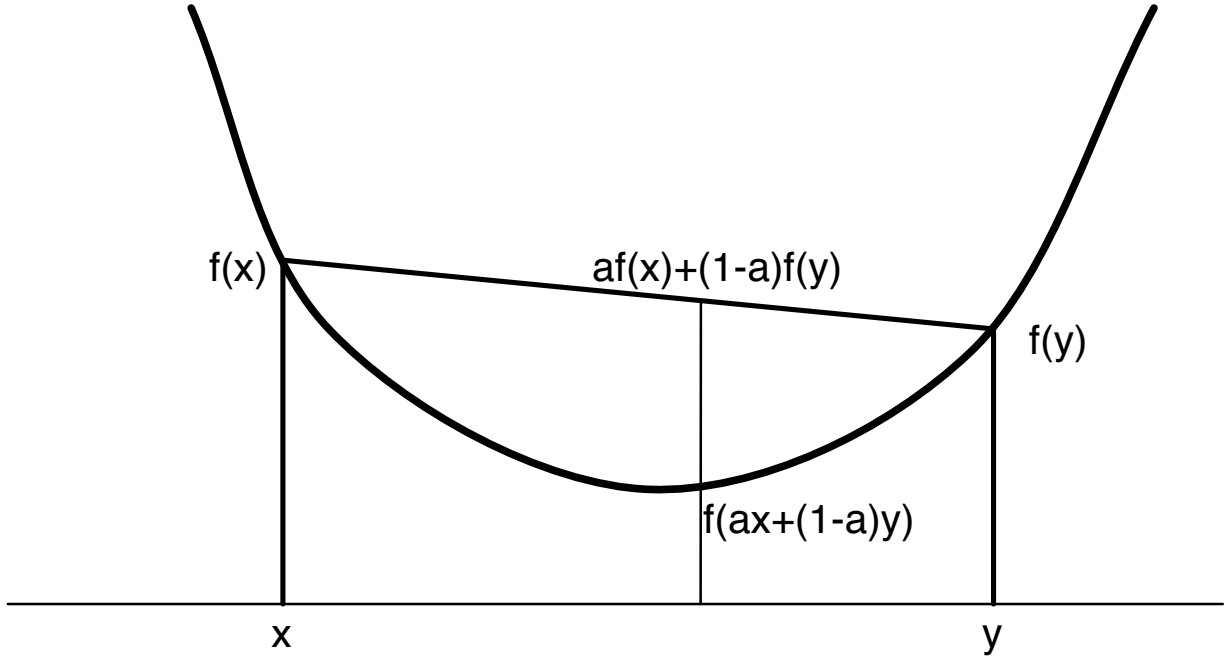
Term 1, Autumn 2011

Convexity

A convex function $f : X \to \mathbb{R}$ is one where

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y)$$

for any $x, y \in X$ and $0 \leq \alpha \leq 1$.



Convex functions have global minimum (unless not bounded below) and there are efficient algorithms to optimize them subject to convex constraints.

Examples: linear programs (LP), quadratic programs (QP), second-order cone programs (SOCP), semi-definite programs (SDP), geometric programs.

Convexity and Approximate Inference

There has been much recent efforts using convex programming techniques to solve inference problems both exactly and approximately.

- Linear programming relaxation as approximate method to find MAP assignment in Markov random fields.
- Attractive Markov random fields: binary case exact and related to a maximum flow-minimum cut problem in graph theory (a linear program). Approximate otherwise.
- Tree-structured convex upper bounds on the log partition function (convexified belief propagation).
- Unified view of approximate inference as optimization on the marginal polytope.
- Learning graphical models using maximum margin principles and convex approximate inference.

...

LP Relaxation for Markov Random Fields

Discrete Markov random fields (MRFs) with pairwise interactions:

$$p(\mathbf{X}) = \frac{1}{Z} \prod_{(ij)} f_{ij}(X_i, X_j) \prod_i f_i(X_i) = \frac{1}{Z} \exp\left(\sum_{(ij)} E_{ij}(X_i, X_j) + \sum_i E_i(X_i) \right)$$

The problem is to find the MAP assignment $\mathbf{X}^{\text{MAP}}$:

$$\mathbf{X}^{\text{MAP}} = \operatorname*{argmax}_{\mathbf{X}} \sum_{(ij)} E_{ij}(X_i, X_j) + \sum_i E_i(X_i)$$

Reformulate in terms of slightly different variables:

$$b_i(x_i) = \delta(X_i = x_i)$$
$$b_{ij}(x_i, x_j) = \delta(X_i = x_i)\delta(X_j = x_j)$$

wher $\delta(\cdot) = 1$ if argument is true, 0 otherwise. Each $b_i(x_i)$ is an indicator for whether variable X_i takes on value x_i. The indicator variables need to satisfy certain constraints:

$b_i(x_i), b_{ij}(x_i, x_j) \in \{0, 1\}$ Indicator variables are binary variables.

$\sum_{x_i} b_i(x_i) = 1$ X_i takes on exactly one value.

$\sum_{x_j} b_{ij}(x_i, x_j) = b_i(x_i)$ Pairwise indicators are consistent with single-site indicators.

LP Relaxation for Markov Random Fields

MAP assignment problem is equivalent to:

$$\underset{\{b_i, b_{ij}\}}{\text{argmax}} \sum_{(ij)} \sum_{x_i, x_j} b_{ij}(x_i, x_j) E_{ij}(x_i, x_j) + \sum_i \sum_{x_i} b_i(x_i) E_i(x_i)$$

with constraints:

$$\forall i, j, x_i, x_j : \quad b_i(x_i), b_{ij}(x_i, x_j) \in \{0, 1\} \quad \sum_{x_i} b_i(x_i) = 1 \quad \sum_{x_j} b_{ij}(x_i, x_j) = b_i(x_i)$$

The linear programming relaxation for MRFs is:

$$\underset{\{b_i, b_{ij}\}}{\text{argmax}} \sum_{(ij)} \sum_{x_i, x_j} b_{ij}(x_i, x_j) E_{ij}(x_i, x_j) + \sum_i \sum_{x_i} b_i(x_i) E_i(x_i)$$

with constraints:

$$\forall i, j, x_i, x_j : \quad b_i(x_i), b_{ij}(x_i, x_j) \in [0, 1] \quad \sum_{x_i} b_i(x_i) = 1 \quad \sum_{x_j} b_{ij}(x_i, x_j) = b_i(x_i)$$

LP Relaxation for Markov Random Fields

- The LP relaxation is a linear program which can be solved efficiently.
- If the solution is integral, i.e. each $b_i(x_i), b_{ij}(x_i, x_j) \in \{0, 1\}$, then the solution corresponds to **the MAP solution $\mathbf{X}^{\mathbf{MAP}}$**.
- LP relaxation is a zero-temperature version of the Bethe free energy formulation of loopy BP, where the Bethe entropy term can be ignored.
- If the MRF is binary and attractive, then a slightly different reformulation of LP relaxation will **always give the MAP solution**.
- Next: we show how to find the MAP solution directly for binary attractive MRFs using network flow.

Attractive Binary MRFs and Max Flow-Min Cut

Binary MRFs:

$$p(\mathbf{X}) = \frac{1}{Z} \exp\left(\sum_{(ij)} W_{ij} \delta(X_i = X_j) + \sum_i c_i X_i \right)$$

The binary MRF is attractive if $W_{ij} \geq 0$ for all i, j. Neighbouring variables prefer to be in the same state in such MRFs.

No loss of generality; can be equivalently expressed as Boltzmann machines with positive interactions.

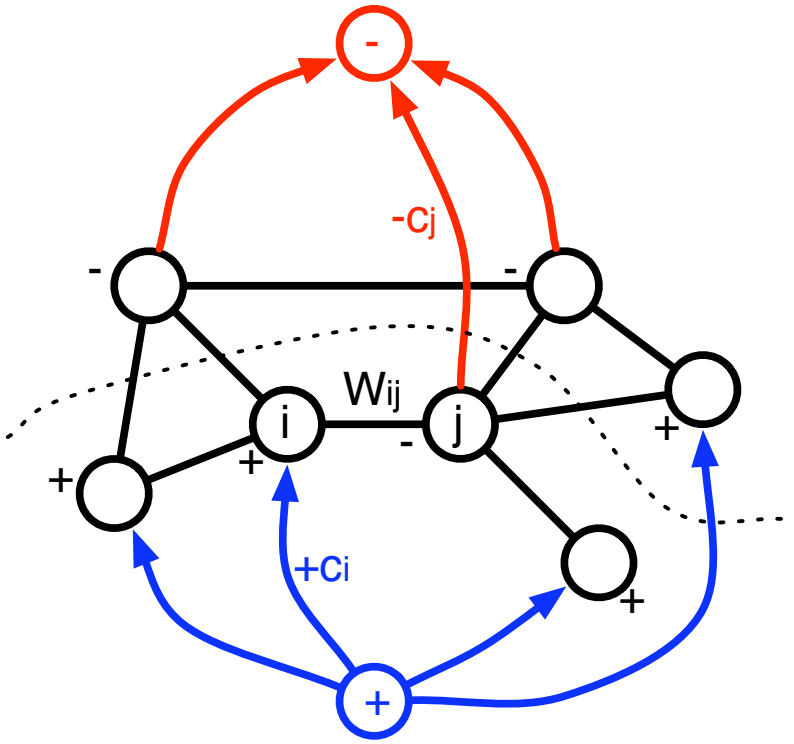
Many practical MRFs are attractive, e.g. image segmentation, webpage classification.

MAP $\mathbf{X}$ can be found efficiently by converting problem into a maximum flow-minimum cut program.

Attractive Binary MRFs and Max Flow-Min Cut

The MAP problem:

$$\underset{\mathbf{x}}{\text{argmax}} \sum_{(ij)} W_{ij}\delta(x_i = x_j) + \sum_i c_i x_i$$



Construct a network as follows:

1. Edges (ij) are undirected with weight $\lambda_{ij} = W_{ij}$;
2. Add a source s and a sink t node;
3. $c_i > 0$: Connect the source node to variable i with weight $\lambda_{si} = c_i$;
4. $c_j < 0$: Connect variable j to the sink node with weight $\lambda_{jt} = -c_j$.

A **cut** is a partition of the nodes into S and T with $s \in S$ and $t \in T$. The weight of the cut is

$$\Lambda(S,T) = \sum_{i \in S, j \in T} \lambda_{ij}$$

The **minimum cut** problem is to find the cut with minimum weight.

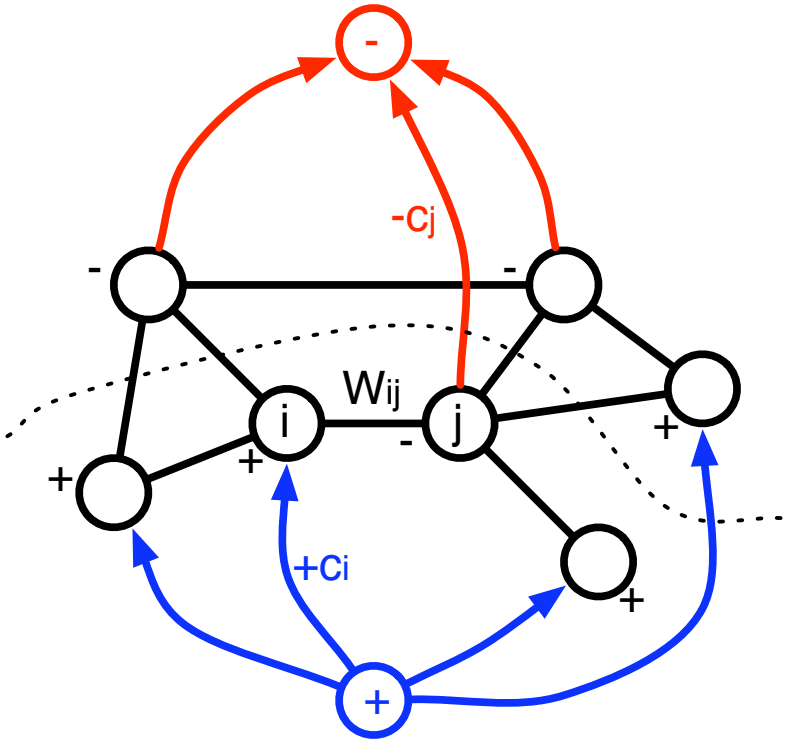
Attractive Binary MRFs and Max Flow-Min Cut

Identify an assignment $\mathbf{X} = \mathbf{x}$ with a cut:

$$S = \{s\} \cup \{i : x_i = 1\}$$
$$T = \{t\} \cup \{j : x_j = 0\}$$

The weight of the cut is:

$$\begin{aligned}\Lambda(S,T) &= \sum_{(ij)} W_{ij}\delta(x_i \neq x_j) \\ &\quad + \sum_i (1 - x_i)\max(0, c_i) \\ &\quad + \sum_j x_j \max(0, -c_j) \\ &= -\sum_{(ij)} W_{ij}\delta(x_i = x_j) - \sum_i x_i c_i + \text{constant}\end{aligned}$$



So finding the minimum cut corresponds to finding the MAP assignment.
How do we find the minimum cut? The minimum cut problem is dual to the **maximum flow problem**, i.e. find the maximum flow allowable from the source to the sink through the network. This can be solved extremely efficiently (see wikipedia entry).
The framework can be generalized to general attractive MRFs, but will not be exact anymore.

Convexity and Exponential Families

An exponential family distribution is parametrized by a natural parameter vector θ and equivalent by its mean parameter vector μ.

$$p(X|\theta) = \exp\left(\theta^\top s(X) - \Phi(\theta)\right)$$

where $\Phi(\theta)$ is the log partition function

$$\Phi(\theta) = \log \sum_x \exp\left(\theta^\top s(x)\right)$$

$\Phi(\theta)$ plays an important role in the characterization of the exponential family. For example, it is a moment generating function for the distribution:

$$\frac{\partial}{\partial\theta}\Phi(\theta) = \mathbb{E}_\theta[s(X)] = \mu(\theta) = \mu$$
$$\frac{\partial^2}{\partial\theta^2}\Phi(\theta) = \mathbb{V}_\theta[s(X)]$$

The second derivative is positive semi-definite, so $\Phi(\theta)$ is convex in θ.

Convexity and Exponential Families

The log partition function and the negative entropy are intimately related. We express the negative entropy as a function of the mean parameter:

$$\Psi(\mu) = \mathbb{E}_\theta[\log p(X|\theta)] = \theta^\top \mu - \Phi(\theta)$$
$$\theta^\top \mu = \Phi(\theta) + \Psi(\mu)$$

The KL divergence between two exponential family distributions $p(X|\theta')$ and $p(X|\theta)$ is:

$$\begin{aligned}\mathsf{KL}(p(X|\theta)\|p(X|\theta')) =& \mathsf{KL}(\theta\|\theta') = \mathbb{E}_\theta[\log p(X|\theta) - \log p(X|\theta')] \\ =& -(\theta')^\top \mu + \Phi(\theta') + \Psi(\mu) \geq 0 \\ \Psi(\mu) \geq& (\theta')^\top \mu - \Phi(\theta')\end{aligned}$$

For any pair of mean and natural parameter vectors.
Because the minimum of the KL divergence is zero, and attained at $\theta = \theta'$, we have:

$$\Psi(\mu) = \sup_{\theta'} (\theta')^\top \mu - \Phi(\theta')$$

The construction on the RHS is called the convex dual of $\Phi(\theta)$. For continuous convex functions, the dual of the dual is the original function, thus:

$$\Phi(\theta) = \sup_{\mu'} \theta^\top \mu' - \Psi(\mu')$$

Convexity and Undirected Trees

Pair-wise MRFs can be parametrized as follows:

$$p(\mathbf{X}) = \frac{1}{Z}\prod_i f_i(X)\prod_{(ij)} f_{ij}(X_i, X_j)$$

$$= \exp\left(\sum_i\sum_{x_i}\theta_i(x_i)\delta(X_i = x_i) + \sum_{(ij)}\sum_{x_i,x_j}\theta_{ij}(x_i, x_j)\delta(X_i = x_i)\delta(X_j = x_j) - \Phi(\theta)\right)$$

So MRFs form an exponential family, with natural and mean parameters:

$$\theta = \left[\theta_i(x_i), \theta_{ij}(x_i, x_j)\forall i, j, x_i, x_j\right]$$
$$\mu = \left[p(X_i = x_i), p(X_i = x_i, X_j = x_j)\forall i, j, x_i, x_j\right]$$

If the MRF has tree structure T, the negative entropy is composed of single-site entropies and mutual informations on edges:

$$\Psi(\mu_T) = \mathbb{E}_{\theta_T}\left[\log\prod_i p(X_i)\prod_{(ij)\in T}\frac{p(X_i, X_j)}{p(X_i)p(X_j)}\right]$$
$$= -\sum_i H(X_i) + \sum_{(ij)\in T} I(X_i, X_j)$$

Convex Upper Bounds on the Log Partition Function

Let us try to upper bound $\Phi(\theta)$.

Imagine a set of spanning trees T for the MRF, each with its own parameters θ_T, μ_T. By padding entries of off-tree edges with zero, we can assume that θ_T has the same dimensionality as θ.

Suppose also that we have a distribution β over the spanning trees so that $\mathbb{E}_\beta[\theta_T] = \theta$. Then by the convexity of $\Phi(\theta)$,

$$\Phi(\theta) = \Phi(\mathbb{E}_\beta[\theta_T]) \leq \mathbb{E}_\beta[\Phi(\theta_T)]$$

Optimizing over all θ_T, we get:

$$\Phi(\theta) \leq \inf_{\theta_T : \mathbb{E}_\beta[\theta_T] = \theta} \mathbb{E}_\beta[\Phi(\theta_T)]$$

Convex Upper Bounds on the Log Partition Function

$$\Phi(\theta) \leq \inf_{\theta_T : \mathbb{E}_\beta[\theta_T] = \theta} \mathbb{E}_\beta[\Phi(\theta_T)]$$

We solve this constrained optimization problem using Lagrange multipliers:

$$\mathcal{L} = \mathbb{E}_\beta[\Phi(\theta_T)] - \mu^\top (\mathbb{E}_\beta[\theta_T] - \theta)$$

Setting the derivatives wrt θ_T to zero, we get:

$$\beta(T)\mu_T - \beta(T)\Pi_T(\mu) = 0$$
$$\mu_T = \Pi_T(\mu)$$

where $\Pi_T(\mu)$ are the Lagrange multipliers corresponding to vertices and edges on the tree T.

Although there can be many θ_T parameters, at optimum they are all constrained: their corresponding mean parameters are all consistent with each other and with μ.

Convex Upper Bounds on the Log Partition Function

$$
\begin{aligned}
\Phi(\theta) &\leq \sup_{\mu} \inf_{\theta_T} \mathbb{E}_\beta[\Phi(\theta_T)] - \mu^\top(\mathbb{E}_\beta[\theta_T] - \theta) \\
&= \sup_{\mu} \mu^\top\theta + \mathbb{E}_\beta[\inf_{\theta_T} \Phi(\theta_T) - \theta_T^\top \Pi_T(\mu)] \\
&= \sup_{\mu} \mu^\top\theta + \mathbb{E}_\beta[-\Psi(\Pi_T(\mu))] \\
&= \sup_{\mu} \mu^\top\theta + \mathbb{E}_\beta\left[\sum_i H_\mu(X_i) - \sum_{(ij)\in T} I_\mu(X_i, X_j)\right] \\
&= \sup_{\mu} \mu^\top\theta + \sum_i H_\mu(X_i) - \sum_{(ij)} \beta_{ij} I_\mu(X_i, X_j)
\end{aligned}
$$

This is a **convexified** Bethe free energy.

References

- **Graphical Models, Exponential Families, and Variational Inference. Wainwright and Jordan. Foundations and Trends in Machine Learning, 2008 1:1-305.**
- Exact Maximum A Posteriori Estimation for Binary Images. Greig, Porteous and Seheult, Journal of the Royal Statistical Society B, 51(2):271-279, 1989.
- Fast Approximate Energy Minimization via Graph Cuts. Boykov, Veksler and Zabih, International Conference on Computer Vision 1999.
- MAP estimation via agreement on (hyper)trees: Message-passing and linear-programming approaches. Wainwright, Jaakkola and Willsky, IEEE Transactions on Information Theory, 2005, 51(11):3697-3717.
- Learning Associative Markov Networks. Taskar, Chatalbashev and Koller, International Conference on Machine Learning, 2004.
- A New Class of Upper Bounds on the Log Partition Function. Wainwright, Jaakkola and Willsky. IEEE Transactions on Information Theory, 2005, 51(7):2313-2335.
- MAP Estimation, Linear Programming and Belief Propagation with Convex Free Energies. Weiss, Yanover and Meltzer, Uncertainty in Artificial Intelligence, 2007.

References