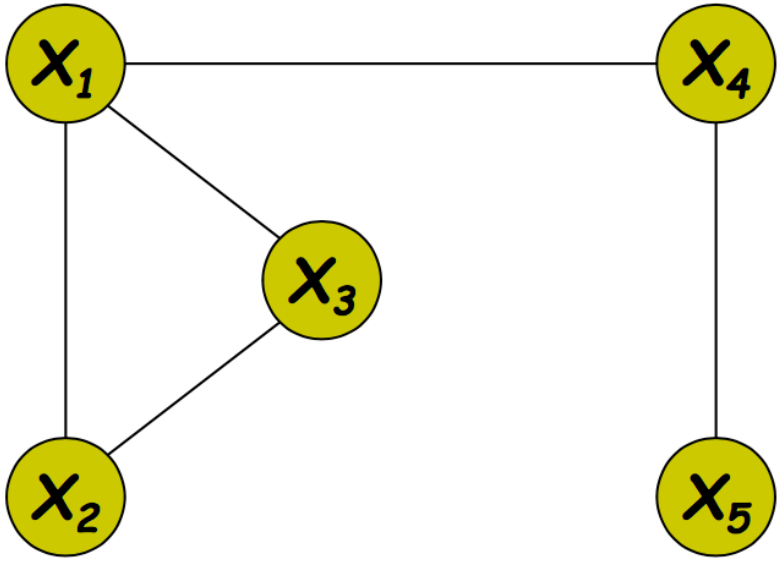
CS839:
Probabilistic Graphical Models

Lecture 8: Learning Fully Observed Undirected Graphical Models

Theo Rekatsinas

Recall: Undirected Graphical Models

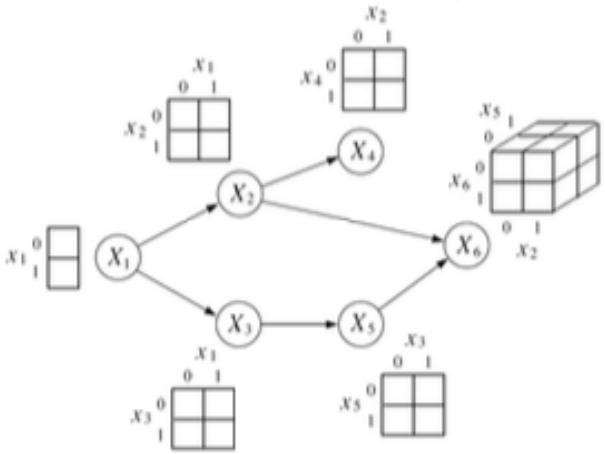


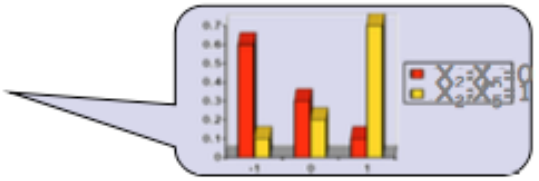
- Pairwise (non-causal) relationships
- We can write down the model, score specific configurations of the RVs but not generate samples
- Contingency constraints on node configurations

Recall: MLE for BNs

- If we assume the parameters for each CPD are globally independent, and all nodes are fully observed, then the log-likelihood function decomposes into a sum of local terms, one per node

$$\ell(\theta; D) = \log p(D \mid \theta) = \log \prod_n \left(\prod_i p(x_{n,i} \mid \mathbf{x}_{n,\pi_i}, \theta_i) \right) = \sum_i \left(\sum_n \log p(x_{n,i} \mid \mathbf{x}_{n,\pi_i}, \theta_i) \right)$$





$$\theta_{ijk}^{ML} = \frac{n_{ijk}}{\sum_{i,j',k} n_{ij'k}}$$

- MLE-based parameter estimation of GM reduces to local est. of each GLIM.

MLE for Undirected Graphical Models

- For **directed** models, the log-likelihood decomposes into a sum of terms, one per family (node plus parents).
- For undirected models, the log-likelihood does not decompose, because the normalization constant Z is a function of **all** parameters.

$$P(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{c \in C} \psi_c(\mathbf{x}_c) \qquad Z = \sum_{x_1, \ldots, x_n} \prod_{c \in C} \psi_c(\mathbf{x}_c)$$

- In general, we need to do inference to learn parameters for undirected models, even in the fully observed case.

Log likelihood for Undirected Graphical Models with tabular clique potentials

- Sufficient statistics: for an MRF (V, E) the number of times that a configuration **x** is observed in a dataset *D can be represented as follows.*

$$m(\mathbf{x}) \overset{\text{def}}{=} \sum_n \delta(\mathbf{x}, \mathbf{x}_n) \quad \text{(total count)}, \quad \text{and} \quad m(\mathbf{x}_c) \overset{\text{def}}{=} \sum_{\mathbf{x}_{V \setminus c}} m(\mathbf{x}) \quad \text{(clique count)}$$

- The log-likelihood is

$$p(D|\theta) = \prod_n \prod_{\mathbf{x}} p(\mathbf{x}|\theta)^{\delta(\mathbf{x},\mathbf{x}_n)}$$

$$\log p(D|\theta) = \sum_n \sum_{\mathbf{x}} \delta(\mathbf{x}, \mathbf{x}_n) \log p(\mathbf{x}|\theta) = \sum_{\mathbf{x}} \sum_n \delta(\mathbf{x}, \mathbf{x}_n) \log p(\mathbf{x}|\theta)$$

$$\ell = \sum_{\mathbf{x}} m(\mathbf{x}) \log \left(\frac{1}{Z} \prod_c \psi_c(\mathbf{x}_c) \right)$$

$$= \sum_c \sum_{\mathbf{x}_c} m(\mathbf{x}_c) \log \psi_c(\mathbf{x}_c) - N \log Z$$

Log likelihood for Undirected Graphical Models with tabular clique potentials

- Sufficient statistics: for an MRF (V, E) the number of times that a configuration **x** is observed in a dataset *D can be represented as follows.*

$$m(\mathbf{x}) \stackrel{\text{def}}{=} \sum_n \delta(\mathbf{x}, \mathbf{x}_n) \quad \text{(total count)}, \quad \text{and} \quad m(\mathbf{x}_c) \stackrel{\text{def}}{=} \sum_{\mathbf{x}_{V \setminus c}} m(\mathbf{x}) \quad \text{(clique count)}$$

- In terms of the counts, the log likelihood is:

$$\log p(D|\theta) = \sum_c \sum_{\mathbf{x}_c} m(\mathbf{x}_c) \log \psi_c(\mathbf{x}_c) - N \log Z$$

Taking the derivative

- Log-likelihood $$\log p(D|\theta) = \sum_c \sum_{\mathbf{x}_c} m(\mathbf{x}_c) \log \psi_c(\mathbf{x}_c) - N \log Z$$
- Fist term: $$\frac{\partial \ell_1}{\partial \psi_c(\mathbf{x}_c)} = m(\mathbf{x}_c) / \psi_c(\mathbf{x}_c)$$
- Second term:

$$\frac{\partial \log Z}{\partial \psi_c(\mathbf{x}_c)} = \frac{1}{Z} \frac{\partial}{\partial \psi_c(\mathbf{x}_c)} \left(\sum_{\tilde{\mathbf{x}}} \prod_d \psi_d(\tilde{\mathbf{x}}_d) \right)$$

$$= \frac{1}{Z} \sum_{\tilde{\mathbf{x}}} \delta(\tilde{\mathbf{x}}_c, \mathbf{x}_c) \frac{\partial}{\partial \psi_c(\mathbf{x}_c)} \left(\prod_d \psi_d(\tilde{\mathbf{x}}_d) \right)$$

$$= \sum_{\tilde{\mathbf{x}}} \delta(\tilde{\mathbf{x}}_c, \mathbf{x}_c) \frac{1}{\psi_c(\tilde{\mathbf{x}}_c)} \frac{1}{Z} \prod_d \psi_d(\tilde{\mathbf{x}}_d)$$

$$= \frac{1}{\psi_c(\mathbf{x}_c)} \sum_{\tilde{\mathbf{x}}} \delta(\tilde{\mathbf{x}}_c, \mathbf{x}_c) p(\tilde{\mathbf{x}}) = \frac{p(\mathbf{x}_c)}{\psi_c(\mathbf{x}_c)}$$

Taking the derivative

- Derivative of log-likelihood

$$\frac{\partial \ell}{\partial \psi_c(\mathbf{x}_c)} = \frac{m(\mathbf{x}_c)}{\psi_c(\mathbf{x}_c)} - N \frac{p(\mathbf{x}_c)}{\psi_c(\mathbf{x}_c)}$$

- Hence we need that:

$$p^*_{MLE}(\mathbf{x}_c) = \frac{m(\mathbf{x}_c)}{N} \overset{\text{def}}{=} \tilde{p}(\mathbf{x}_c)$$

- This says that:
  - For the maximum likelihood estimates of the parameters, for each clique, the model marginals must be equal to the observed marginals (empirical counts)
  - This is only a condition that the parameters should satisfy!
  - It does not tell us how to get the maximum likelihood estimates.

MLE for Undirected Graphical Models

- Case 1: The model is **decomposable** (triangulated graph) and all the clique potentials are defined on maximal cliques.
  - The MLE of clique potentials are equal to the empirical marginals (or conditionals) of the corresponding clique.
  - Solve MLE by inspection

- Decomposable models
  - G is decomposable, G is triangulated, G has a junction tree

$$p(\mathbf{x}) = \frac{\prod_c \psi_c(\mathbf{x}_c)}{\prod_s \varphi_s(\mathbf{x}_s)}$$

  - Ex.: Chain X1 – X2 – X3

$$p_{MLE}(X1, X2, X3) = \frac{\tilde{p}(X1, X2)\tilde{p}(X2, X3)}{\tilde{p}(X2)}$$

$$p_{MLE}(X1, X2) = \sum_{X3} \tilde{p}(X1, X2, X3) = \tilde{p}(X1|X2) \sum_{X3} \tilde{p}(X2, X3) = \tilde{p}(X1, X2)$$

$$p_{MLE}(X2, X3) = \tilde{p}(X2, X3)$$

MLE for Undirected Graphical Models

- Decomposable models
  - G is decomposable, G is triangulated, G has a junction tree
  - Ex.: Chain X1 – X2 – X3

$$p(\mathbf{x}) = \frac{\prod_c \psi_c(\mathbf{x}_c)}{\prod_s \varphi_s(\mathbf{x}_s)}$$

$$p_{MLE}(X1, X2, X3) = \frac{\tilde{p}(X1, X2)\tilde{p}(X2, X3)}{\tilde{p}(X2)}$$

$$p_{MLE}(X1, X2) = \sum_{X3} \tilde{p}(X1, X2, X3) = \tilde{p}(X1|X2) \sum_{X3} \tilde{p}(X2, X3) = \tilde{p}(X1, X2)$$

$$p_{MLE}(X2, X3) = \tilde{p}(X2, X3)$$

- To compute the clique potentials we just use the empirical marginals (or conditionals), i.e., the separator must be divided into one of its neighbors. Then Z = 1

$$\hat{\psi}_{12}^{MLE}(x_1, x_2) = \tilde{p}(x_1, x_2) \qquad \hat{\psi}_{23}^{MLE}(x_2, x_3) = \frac{\tilde{p}(x_2, x_3)}{\tilde{p}(x_2)} = \tilde{p}(x_2 \mid x_3)$$

MLE for Undirected Graphical Models

- Case 2: The model is **non-decomposable**, the potentials are defined as non-maximal cliques. We cannot equate MLE of clique potentials to empirical marginals (or conditionals)
  - Iterative potential fitting
  - Generalized Iterative Scaling

Iterative Proportional Fitting (IPF)

- From the log-likelihood: $\frac{\partial \ell}{\partial \psi_c(\mathbf{x}_c)} = \frac{m(\mathbf{x}_c)}{\psi_c(\mathbf{x}_c)} - N \frac{p(\mathbf{x}_c)}{\psi_c(\mathbf{x}_c)}$
- Let's rewrite in a different way: $\frac{m(\mathbf{x}_c)}{N\psi_c(\mathbf{x}_c)} = \frac{p(\mathbf{x}_c)}{\psi_c(\mathbf{x}_c)}$ or $\frac{\tilde{p}(\mathbf{x}_c)}{\psi_c(\mathbf{x}_c)} = \frac{p(\mathbf{x}_c)}{\psi_c(\mathbf{x}_c)}$
  - The clique potentials implicitly appear in the model marginal $p(\mathbf{x}_c) = f(\psi_c(\mathbf{x}_c))$
- Let's forget a closed form solution and focus on a **fixed-point iteration** method

$$\frac{\tilde{p}(\mathbf{x}_c)}{\psi_c^{(t+1)}(\mathbf{x}_c)} = \frac{p(\mathbf{x}_c)}{\psi_c^{(t)}(\mathbf{x}_c)} \quad \Rightarrow \quad \psi_c^{(t+1)}(\mathbf{x}_c) = \psi_c^{(t)}(\mathbf{x}_c) \frac{\tilde{p}(\mathbf{x}_c)}{p^{(t)}(\mathbf{x}_c)}$$

- Need to run inference for $p^{(t)}(\mathbf{x}_c)$

Properties of IPF Updates

- Set of fixed-point equations:

$$\psi_c^{(t+1)}(\mathbf{x}_c) = \psi_c^{(t)}(\mathbf{x}_c)\frac{\tilde{p}(\mathbf{x}_c)}{p^{(t)}(\mathbf{x}_c)}$$

- We can show that it is also a coordinate ascent algorithm (coordinates=parameters of clique potentials)

- At each step, it will increase the log-likelihood, and it will converge to a global maximum.

$$\max \ell \Leftrightarrow \min KL\big(\tilde{p}(x)\,\|\,p(x\mid\theta)\big) = \sum_x \tilde{p}(x)\log\frac{\tilde{p}(x)}{p(x\mid\theta)}$$

- Maximizing the log likelihood is equivalent to minimizing the KL divergence (cross entropy)
- The max-entropy principle to parameterization offers a dual perspective to the MLE.

$$\begin{aligned}\min_p \quad & \mathrm{KL}\big(p(x)\,\|\,h(x)\big) \\ & \overset{\text{def}}{=} \sum_x p(x)\log\frac{p(x)}{h(x)} = -\mathrm{H}(p) - \sum_x p(x)\log h(x) \\ \text{s.t.} \quad & \sum_x p(x) f_i(x) = \alpha_i \\ & \sum_x p(x) = 1\end{aligned}$$

MLE for undirected graphical models

- What have we seen so far?

$$p(\mathbf{x}) = \frac{\prod_c \psi_c(\mathbf{x}_c)}{\prod_s \varphi_s(\mathbf{x}_s)}$$

- Decomposable graphs
  - Clique potentials correspond to marginals or conditionals

- Clique potentials that correspond to full tables
  - Iterative Proportional fitting

$$\psi_c^{(t+1)}(\mathbf{x}_c) = \psi_c^{(t)}(\mathbf{x}_c) \frac{\tilde{p}(\mathbf{x}_c)}{p^{(t)}(\mathbf{x}_c)}$$

- What about models that are parameterized more compactly?

$$\psi_c(\mathbf{x}_c) = \exp\left(\sum_c \theta_k f_k(\mathbf{x}_c)\right)$$

Feature-parameterized clique potentials

- So far we saw the most general form of an undirected graphical model: cliques are parameterized by general **tabular** potential functions

- For large cliques these potentials are exponentially costly for inference. Also, we have exponentially many parameters to learn from limited data.

- **Solution: ?**

Feature-parameterized clique potentials

- So far we saw the most general form of an undirected graphical model: cliques are parameterized by general **tabular** potential functions

- For large cliques these potentials are exponentially costly for inference. Also, we have exponentially many parameters to learn from limited data.

- **Solution: Change the graphical model to make cliques smaller.**

Feature-parameterized clique potentials

- So far we saw the most general form of an undirected graphical model: cliques are parameterized by general **tabular** potential functions

- For large cliques these potentials are exponentially costly for inference. Also, we have exponentially many parameters to learn from limited data.

- **Solution: Change the graphical model to make cliques smaller.**

- This changes the dependencies and may force us to make more independence assumptions than what we had

Feature-parameterized clique potentials

- So far we saw the most general form of an undirected graphical model: cliques are parameterized by general **tabular** potential functions

- For large cliques these potentials are exponentially costly for inference. Also, we have exponentially many parameters to learn from limited data.

- **Solution: Keep the same graphical model but use less parameters to define the clique potentials**
  - Recall parameter sharing for BNs

- This is the idea behind feature-based models.

Features

- Let a clique correspond to three consecutive characters
- How would you define p(c1,c2,c3)?

Features

- Let a clique correspond to three consecutive characters
- How would you define p(c1,c2,c3)?
  - For all possible character combinations you need $26^3 - 1$ parameters.
  - But there are sequences that are unlikely: kfd

- A “feature” is a function that is non-zero for a few particular inputs. Think of Boolean features.
  - Is “ing” the input sequence? Then 1 otherwise 0.

- We can define features for continuous features as well.

Features as potentials

- Each feature function can be converted to a potential by taking the exponent of it. We can multiply these potentials together to get a **clique potential.**

- **Example:**

$$\psi_c(c_1,c_2,c_3) = e^{\theta_{\text{ing}} f_{\text{ing}}} \times e^{\theta_{?\text{ed}} f_{?\text{ed}}} \times \ldots$$

$$= \exp\left\{\sum_{k=1}^{K} \theta_k f_k(c_1,c_2,c_3)\right\}$$

- There is still an exponential number of setting but we only use K parameters corresponding to the K features.
  - Can we recover the tabular representation?

Combining Features

- Each feature has a weight θ_k which represents the numerical strength of the feature and whether it increases or decreases the probability of a clique.
- The marginal over the clique is a generalized exponential family distribution (a generalized linear model)

$$p(c_1,c_2,c_3) \propto \exp\left\{\begin{array}{l}\theta_{\text{ing}} f_{\text{ing}}(c_1,c_2,c_3) + \theta_{?\text{ed}} f_{?\text{ed}}(c_1,c_2,c_3) + \\ \theta_{\text{qu?}} f_{\text{qu?}}(c_1,c_2,c_3) + \theta_{\text{zzz}} f_{\text{zzz}}(c_1,c_2,c_3) + \cdots\end{array}\right\}$$

- The features may be overlapping across cliques $\psi_c(\mathbf{x}_c) \stackrel{\text{def}}{=} \exp\left\{\sum_{i \in I_c} \theta_k f_k(\mathbf{x}_{c_i})\right\}$

Feature-based model

- Joint distribution: $p(\mathbf{x}) = \frac{1}{Z(\theta)} \prod_c \psi_c(\mathbf{x}_c) = \frac{1}{Z(\theta)} \exp\left\{ \sum_c \sum_{i \in I_c} \theta_k f_k(\mathbf{x}_{c_i}) \right\}$

- We can use the simplified form $p(\mathbf{x}) = \frac{1}{Z(\theta)} \exp\left\{ \sum_i \theta_i f_i(\mathbf{x}_{c_i}) \right\}$

- The features correspond to the sufficient statistics of our model.

- We need to learn parameters θ_k

Feature-based model

- Joint distribution: $p(\mathbf{x}) = \frac{1}{Z(\theta)} \prod_c \psi_c(\mathbf{x}_c) = \frac{1}{Z(\theta)} \exp\left\{ \sum_c \sum_{i \in I_c} \theta_k f_k(\mathbf{x}_{c_i}) \right\}$
- We can use the simplified form $p(\mathbf{x}) = \frac{1}{Z(\theta)} \exp\left\{ \sum_i \theta_i f_i(\mathbf{x}_{c_i}) \right\}$
- The features correspond to the sufficient statistics of our model.
- We need to learn parameters θ_k
- What about IPF?
  - Not clear how to use this rule to update the parameters and potentials

$$\psi_c^{(t+1)}(\mathbf{x}_c) = \psi_c^{(t)}(\mathbf{x}_c) \frac{\tilde{p}(\mathbf{x}_c)}{p^{(t)}(\mathbf{x}_c)}$$

MLE of Feature-based Undirected Graphical Models

- Objective: scaled likelihood function

$$\tilde{\ell}(\theta;\mathcal{D}) = \ell(\theta;\mathcal{D})/N = \frac{1}{N}\sum_n \log p(x_n \mid \theta)$$
$$= \sum_x \tilde{p}(x) \log p(x \mid \theta)$$
$$= \sum_x \tilde{p}(x) \sum_i \theta_i f_i(x) - \log Z(\theta)$$

- Main difficulties: the partition function is a complex function of the parameters. If we take a derivative Z appears in the denominator. Nothing changes. We want to avoid computing Z.

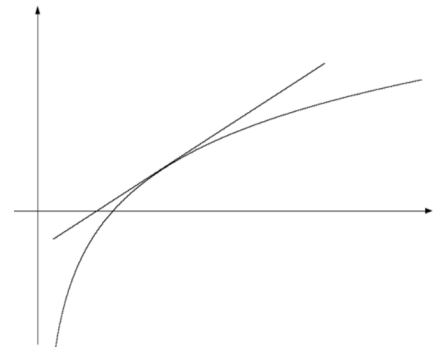
- Approximation time…

MLE of Feature-based Undirected Graphical Models

- Objective: scaled likelihood function

$$\tilde{\ell}(\theta;D) = \ell(\theta;D)/N = \frac{1}{N}\sum_n \log p(x_n \mid \theta)$$
$$= \sum_x \tilde{p}(x) \log p(x \mid \theta)$$
$$= \sum_x \tilde{p}(x) \sum_i \theta_i f_i(x) - \log Z(\theta)$$

- We replace logZ by its upper bound $\log Z(\theta) <= \mu Z(\theta) - \log\mu - 1$ where $\mu = Z^{-1}(\theta^{(t)})$



- Thus we have

$$\tilde{\ell}(\theta;D) \geq \sum_x \tilde{p}(x) \sum_i \theta_i f_i(x) - \frac{Z(\theta)}{Z(\theta^{(t)})} - \log Z(\theta^{(t)}) + 1$$

MLE of Feature-based Undirected Graphical Models

- We have $$\tilde{\ell}(\theta;D) \geq \sum_x \tilde{p}(x) \sum_i \theta_i f_i(x) - \frac{Z(\theta)}{Z(\theta^{(t)})} - \log Z(\theta^{(t)}) + 1$$
- We define $$\Delta\theta_i^{(t)} \stackrel{\text{def}}{=} \theta_i - \theta_i^{(t)}$$

$$\begin{aligned}
\tilde{\ell}(\theta;D) &\geq \sum_x \tilde{p}(x) \sum_i \theta_i f_i(x) - \frac{1}{Z(\theta^{(t)})} \sum_x \exp\left\{\sum_i \theta_i f_i(x)\right\} - \log Z(\theta^{(t)}) + 1 \\
&= \sum_i \theta_i \sum_x \tilde{p}(x) f_i(x) - \frac{1}{Z(\theta^{(t)})} \sum_x \exp\left\{\sum_i \theta_i^{(t)} f_i(x)\right\} \exp\left\{\sum_i \Delta\theta_i^{(t)} f_i(x)\right\} - \log Z(\theta^{(t)}) + 1 \\
&= \sum_i \theta_i \sum_x \tilde{p}(x) f_i(x) - \sum_x p(x \mid \theta^{(t)}) \exp\left\{\sum_i \Delta\theta_i^{(t)} f_i(x)\right\} - \log Z(\theta^{(t)}) + 1
\end{aligned}$$

- We assume $f_i(x) \geq 0, \sum_i f_i = 1$. Also by convexity of exp $\exp(\sum_i \pi_i \mathbf{x}_i) \leq \sum_i \pi_i \exp(\mathbf{x_i})$

$$\tilde{\ell}(\theta;D) \geq \sum_i \theta_i \sum_x \tilde{p}(x) f_i(x) - \sum_x p(x \mid \theta^{(t)}) \sum_i f_i(x) \exp\left(\Delta\theta_i^{(t)}\right) - \log Z(\theta^{(t)}) + 1 \stackrel{\text{def}}{=} \Lambda(\theta)$$

MLE of Feature-based Undirected Graphical Models

- We have $\tilde{\ell}(\theta;D) \geq \sum_i \theta_i \sum_x \tilde{p}(x) f_i(x) - \sum_x p(x \mid \theta^{(t)}) \sum_i f_i(x) \exp\left(\Delta\theta_i^{(t)}\right) - \log Z(\theta^{(t)}) + 1 \stackrel{\text{def}}{=} \Lambda(\theta)$

- We take the derivative
  - $p^{(t)}(x)$ is the unnormalized version of $p(x|\theta^{(t)})$

$$\frac{\partial \Lambda}{\partial \theta_i} = \sum_x \tilde{p}(x) f_i(x) - \exp\left(\Delta\theta_i^{(t)}\right) \sum_x p(x \mid \theta^{(t)}) f_i(x)$$

$$e^{\Delta\theta_i^{(t)}} = \frac{\sum_x \tilde{p}(x) f_i(x)}{\sum_x p(x \mid \theta^{(t)}) f_i(x)} = \frac{\sum_x \tilde{p}(x) f_i(x)}{\sum_x p^{(t)}(x) f_i(x)} Z(\theta^{(t)})$$

- Our updates are:

$$\theta_i^{(t+1)} = \theta_i^{(t)} + \Delta\theta_i^{(t)} \Rightarrow p^{(t+1)}(x) = p^{(t)}(x) \prod_i e^{\Delta\theta_i^{(t)} f_i(x)}$$

$$p^{(t+1)}(x) = \frac{p^{(t)}(x)}{Z(\theta^{(t)})} \prod_i \left(\frac{\sum_x \tilde{p}(x) f_i(x)}{\sum_x p^{(t)}(x) f_i(x)} Z(\theta^{(t)}) \right)^{f_i(x)}$$

$$= \frac{p^{(t)}(x)}{Z(\theta^{(t)})} \prod_i \left(\frac{\sum_x \tilde{p}(x) f_i(x)}{\sum_x p^{(t)}(x) f_i(x)} \right)^{f_i(x)} \left(Z(\theta^{(t)}) \right)^{\sum_i f_i(x)}$$

$$= p^{(t)}(x) \prod_i \left(\frac{\sum_x \tilde{p}(x) f_i(x)}{\sum_x p^{(t)}(x) f_i(x)} \right)^{f_i(x)}$$

Summary

- Iterative Proportional Fitting (IPF) is a general algorithm for MLE of UGMs
  - A **fixed-point equation** for potentials over single cliques, uses coordinate ascent
  - Requires the potential to be fully parameterized
  - The clique described by the potentials does not have to be max-clique
  - For fully decomposable model, reduces to a single step iteration
- Generalized Iterative Scaling (GIS)
  - Iterative scaling on general UGM with feature-based potentials
  - IPF is a special case of GIS where the clique potential is built on features defined as indicator functions of the clique configurations.

Summary

GIS:

$$p^{(t+1)}(x) = p^{(t)}(x) \prod_i \left(\frac{\sum_x \tilde{p}(x) f_i(x)}{\sum_x p^{(t)}(x) f_i(x)} \right)^{f_i(x)}$$

$$\theta_i^{(t+1)} = \theta_i^{(t)} + \log \left(\frac{\sum_x \tilde{p}(x) f_i(x)}{\sum_x p^{(t)}(x) f_i(x)} \right)$$

IPF:

$$\psi_c^{(t+1)}(\mathbf{x}_c) = \psi_c^{(t)}(\mathbf{x}_c) \frac{\tilde{p}(\mathbf{x}_c)}{p^{(t)}(\mathbf{x}_c)}$$