Georgia Institute of Technology

CSIP Center for Signal & Image Processing

ECE 8803

Parameter Learning in Graphical Models

Module 12
Bayesian Networks:

1. known Structure &
2. Partially Observed Variables

Faramarz Fekri

Center for Signal and Information Processing

Overview

- Learning parameters in Graphical Models (BN):
  - Partially Observed variables and known Structure:
    - Decoupling of Observation Mechanism
    - Likelihood for Fully/Partially Observed Variables
    - MLE for parameter estimation in Partially Observed Data
      - Gradient Method
      - Expectation Maximization (EM)
        - » Example BN
        - » General BN

Chapter 19 in Koller

Unobserved Variables

- A variable can be unobserved (latent, hidden, missing) because:
  - It is an imaginary quantity meant to provide some simplified and abstract view of the data generation process
    - Eg. Mixture models, topic modeling, image context
  - It is a real-world object and/or phenomena, but difficult or impossible to measure
    - Eg. Causes of disease, evolutionary ancestor
  - It is a real-world object and/or phenomena, but sometimes wasn't measured, because of faulty sensors etc.
- Discrete latent variables can be used to partition/cluster data into subgroups
- Continuous latent variables (factors) can be used for dimensionality reduction (factor analysis, etc.)

Decoupling of Observation Mechanism

- To solve the problem, we will ignore the "missing data mechanism (e.g., random, deliberate, biased toward certain outcomes, etc)" and focus only on the likelihood.
  - Question: when the above assumption is fine?
    - Missing at Random (MAR): The probability that the value of X_i is missing is independent of its actual value, given other observed values.

Assuming missing at random, we now need to understand how does missing data affect the likelihood function

- **likelihood function:**

$$L(D:\theta) = P(D|\Theta) = \prod_{m} P(o[m]|\Theta)$$

Observed variables

- **It turns out that dealing with this likelihood function when variables is partially observed is considerably more difficult than when fully observed case.**

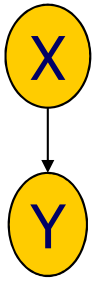
Likelihood for Fully Observed Variables

Example: Two-node scenario

Data D:

X	Y
x^0	y^0
x^0	y^1
x^1	y^0

P(X)	
x^0	x^1
θ_{x0}	θ_{x1}



- **likelihood function:**

$$
\begin{aligned}
L(D:\theta) &= P(x[1],y[1])\cdot P(x[2],y[2])\cdot P(x[3],y[3]) \\
&= P(x^0,y^0)\cdot P(x^0,y^1)\cdot P(x^1,y^0) \\
&= \theta_{x^0}\cdot\theta_{y^0|x^0}\cdot\theta_{x^0}\cdot\theta_{y^1|x^0}\cdot\theta_{x^1}\cdot\theta_{y^0|x^1} \\
&= \left(\theta_{x^0}\cdot\theta_{x^0}\cdot\theta_{x^1}\right)\cdot\left(\theta_{y^0|x^0}\cdot\theta_{y^1|x^0}\right)\cdot\left(\theta_{y^0|x^1}\right)
\end{aligned}
$$

	P(Y/X)	
X	y^0	y^1
x^0	$\theta_{y0\|x0}$	$\theta_{y1\|x0}$
x^1	$\theta_{y0\|x1}$	$\theta_{y1\|x1}$

1. Likelihood decomposes by variables
2. Likelihood decomposes within CPDs
3. Log of Likelihood function is concave → unique global maximum that has a simple analytic closed form.

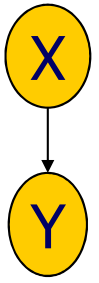
Likelihood for Partially Observed Variables

Example: Two-node scenario

Data D:

X	Y
?	y^0
x^0	y^1
?	y^0

P(X)	
x^0	x^1
θ_{x0}	θ_{x1}



X	P(Y/X)	
	y^0	y^1
x^0	$\theta_{y0\|x0}$	$\theta_{y1\|x0}$
x^1	$\theta_{y0\|x1}$	$\theta_{y1\|x1}$

- **likelihood function:**

$$
\begin{aligned}
L(D:\theta) &= P(y^0)\cdot P(x^0,y^1)\cdot P(y^0) \\
&= \left(\sum_{x\in X} P(x,y^0)\right)\cdot P(x^0,y^1)\cdot\left(\sum_{x\in X} P(x,y^0)\right) \\
&= \left(\theta_{x^0}\cdot\theta_{y^0|x^0}+\theta_{x^1}\cdot\theta_{y^0|x^1}\right)\cdot\theta_{x^0}\cdot\theta_{y^1|x^0}\cdot\left(\theta_{x^0}\cdot\theta_{y^0|x^0}+\theta_{x^1}\cdot\theta_{y^0|x^1}\right) \\
&== \left(\theta_{x^0}\cdot\theta_{y^0|x^0}+\theta_{x^1}\cdot\theta_{y^0|x^1}\right)^2\cdot\theta_{x^0}\cdot\theta_{y^1|x^0}
\end{aligned}
$$

1. Likelihood does not decompose by variables
2. Likelihood does not decompose within CPDs
3. Computing likelihood per instance requires inference!

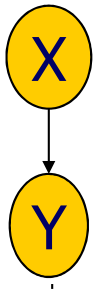
Likelihood for Partially Observed Variables (II)

Again: Two-node scenario, binary r.v X and Y

- **likelihood function:** $L(\theta, D) = P(D \mid \theta) = \prod_{m=1}^{M} P(O[m] \mid \theta)$

(Assume fully observed): $= \theta_{x^1}^{M(x^1)} \theta_{x^0}^{M(x0)} \theta_{y^1|x^1}^{M(y^1,x^1)} \theta_{y^1|x^0}^{M(y^1,x0)} \theta_{y^0|x^0}^{M(y^0,x^0)} \theta_{y^0|x^1}^{M(y^0,x^1)}$

P(X)	
x^0	x^1
θ_{x0}	θ_{x1}



Data D: total of 43 pairs (X,Y)

- Fully observed: $L(\theta, D) = \theta_{x^1}^{29} \; \theta_{x^0}^{14} \; \theta_{y^1|x^1}^{13} \; \theta_{y^1|x^0}^{10} \; \theta_{y^0|x^0}^{4} \; \theta_{y^0|x^1}^{16}$

- Partial observation: Now assume that the first data sample, instead of being (x^0, y^1), is a partial observation (?, y^1):
  - Two choices for Likelihood function:
  - One choice corresponds to (?=x^0, y^1):

$$L(\theta, D) = \theta_{x^1}^{29} \; \theta_{x^0}^{14} \; \theta_{y^1|x^1}^{13} \; \theta_{y^1|x^0}^{10} \; \theta_{y^0|x^0}^{4} \; \theta_{y^0|x^1}^{16}$$

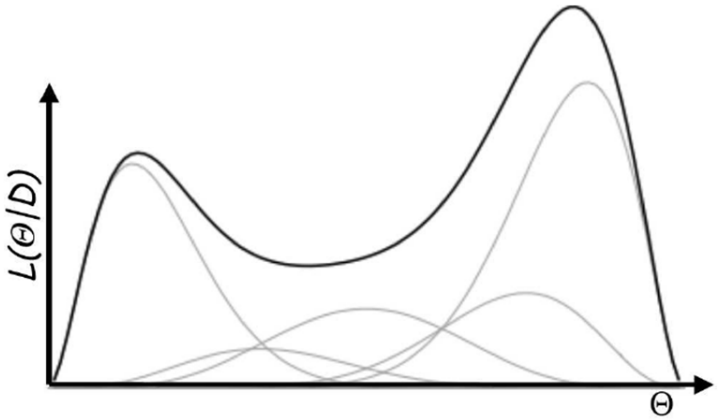
  - Other choice corresponds to (?=x^1, y^1):

$$L(\theta, D) = \theta_{x^1}^{30} \; \theta_{x^0}^{13} \; \theta_{y^1|x^1}^{14} \; \theta_{y^1|x^0}^{9} \; \theta_{y^0|x^0}^{4} \; \theta_{y^0|x^1}^{16}$$

X	P(Y/X)	
	y^0	y^1
x^0	$\theta_{y0\|x0}$	$\theta_{y1\|x0}$
x^1	$\theta_{y0\|x1}$	$\theta_{y1\|x1}$

Likelihood for Partially Observed Variables (III)

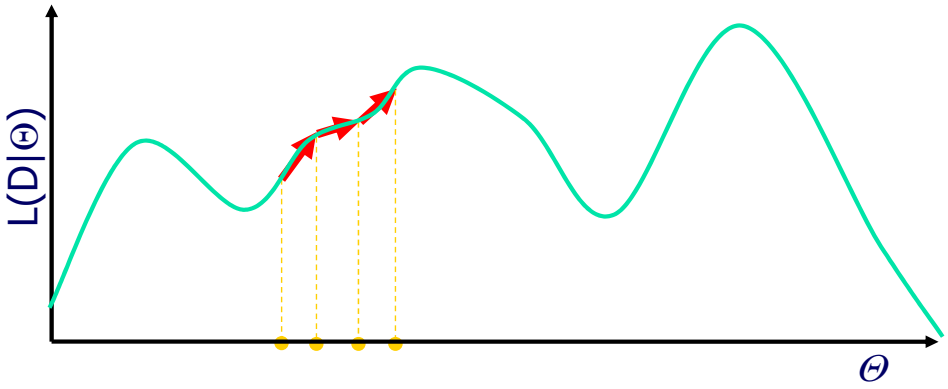
- If we had L partially observed samples in data D, assuming binary case in the previous example, we would get 2^L possible likelihood functions:
  - Adding the likelihood functions (e.g, 2^L picks):



- Log Likelihood function is a multimodal function→ multiple local optima, hence, harder to optimize!
- Also, it may have multiple global optima (Thus, even if we solve the optimization problem perfectly and find the global optima, the problem is unidentifiable (i.e., cannot uniquely identify the model parameters)
- Also, no longer decomposable likelihood function (i.e., model parameters are coupled)

MLE for Partially Observed Data

- Two general ways to solve for model parameters that maximize the multimodal likelihood function:



1. Standard gradient (ascent) method (we will not cover)
   - Taking steps proportional to the direction of steepest gradient of likelihood w.r.t. parameters)

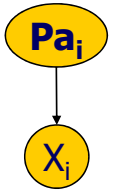
Theorem:
(Page 864 Koller)

$$\frac{\partial \log P(D \mid \Theta)}{\partial \theta_{x_i, pa_i}} = \frac{1}{\theta_{x_i, pa_i}} \sum_m P(x_i, pa_i \mid o[m], \Theta)$$

each training sample

Observed data in m^{th} sample

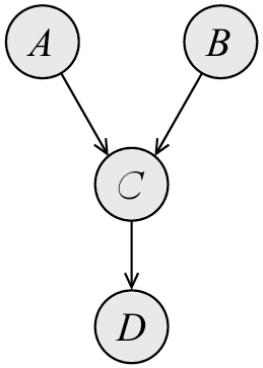
Pa_i	X_i H	X_i T
(H,…,H)	$\Theta_{H\mid(H,\ldots,H)}$	$\Theta_{T\mid(H,\ldots,H)}$
(H,…,T)	$\Theta_{T\mid(H,\ldots,T)}$	$\Theta_{T\mid(H,\ldots,T)}$



   - Requires computation of $P(X_i, pa_i \mid o[m], \Theta)$ for all nodes Xi, and data sample m.
   - For reasonable convergence, must be combined with advanced methods (conjugate gradient, line search)
2. Expectation Maximization (EM)

Example: Gradient Method

Example: Consider the network shown and a partially specified data case $o = \langle a^1, ?, ?, d^0 \rangle$.



- Four possible completions (assume binary case): a^1, b^0, c^0, d^0
  a^1, b^0, c^1, d^0
  a^1, b^1, c^0, d^0
  a^1, b^1, c^1, d^0

Assume that our current $\boldsymbol{\theta}$ is:

$$
\begin{aligned}
\boldsymbol{\theta}_{a^1} &= 0.3 \\
\boldsymbol{\theta}_{b^1} &= 0.9 \\
\boldsymbol{\theta}_{c^1|a^0,b^0} &= 0.83 \\
\boldsymbol{\theta}_{c^1|a^0,b^1} &= 0.09 \\
\boldsymbol{\theta}_{c^1|a^1,b^0} &= 0.6 \\
\boldsymbol{\theta}_{c^1|a^1,b^1} &= 0.2 \\
\boldsymbol{\theta}_{d^1|c^0} &= 0.1 \\
\boldsymbol{\theta}_{d^1|c^1} &= 0.8.
\end{aligned}
$$

$$
\begin{aligned}
P(\langle a^1, b^1, c^1, d^0 \rangle) &= 0.3 \cdot 0.9 \cdot 0.2 \cdot 0.2 = 0.0108 \\
P(\langle a^1, b^1, c^0, d^0 \rangle) &= 0.3 \cdot 0.9 \cdot 0.8 \cdot 0.9 = 0.1944 \\
P(\langle a^1, b^0, c^1, d^0 \rangle) &= 0.3 \cdot 0.1 \cdot 0.6 \cdot 0.2 = 0.0036 \\
P(\langle a^1, b^0, c^0, d^0 \rangle) &= 0.3 \cdot 0.1 \cdot 0.4 \cdot 0.9 = 0.0108.
\end{aligned}
$$

To compute the posterior probability of these instances given the partial observation o, we divide the probability of each instance with the total probability (sum over c and d), $P(o)=p(a^1, d^0)=0.2196$, we get:

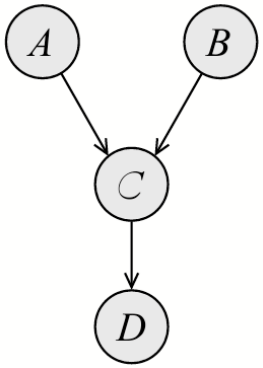
$$
\begin{aligned}
P(\langle a^1, b^1, c^1, d^0 \rangle \mid \boldsymbol{o}) &= 0.0492 \\
P(\langle a^1, b^1, c^0, d^0 \rangle \mid \boldsymbol{o}) &= 0.8852 \\
P(\langle a^1, b^0, c^1, d^0 \rangle \mid \boldsymbol{o}) &= 0.0164 \\
P(\langle a^1, b^0, c^0, d^0 \rangle \mid \boldsymbol{o}) &= 0.0492.
\end{aligned}
$$

Gradient Example: Updating Parameter $\theta_{d^0|c^0}$

$$\frac{\partial \log P(D|\Theta)}{\partial \theta_{x_i,pa_i}} = \frac{1}{\theta_{x_i,pa_i}} \sum_m P(x_i, pa_i | o[m], \Theta)$$

For each observed data

$$\frac{\partial \ell(\boldsymbol{\theta} : \mathcal{D})}{\partial P(d^0 \mid c^0)} = ?$$



$$\frac{P(d^0, c^0 \mid \boldsymbol{o})}{P(d^0 \mid c^0)} = \frac{0.8852 + 0.0492}{0.9} = 1.0382 \quad \Leftarrow \quad o = \langle a^1, ?, ?, d^0 \rangle$$

Another sample: $\boldsymbol{o}' = \langle a^0, ?, ?, d^1 \rangle$.

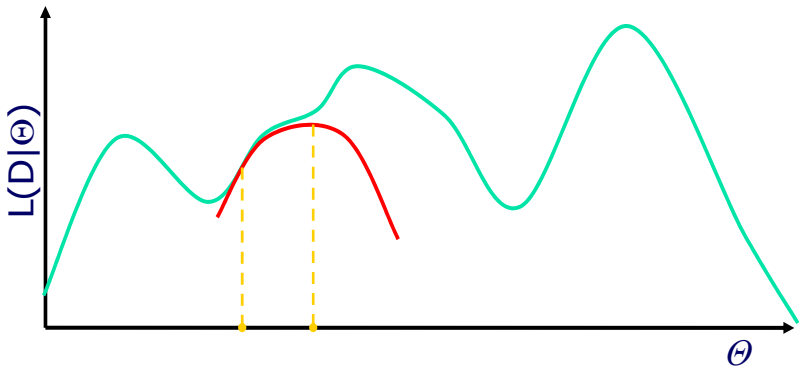
$$\frac{P(d^0, c^0 \mid \boldsymbol{o}')}{P(d^0 \mid c^0)} = \frac{0}{0.9} = 0$$

$\Rightarrow$ Assuming only two sample data, Need to sum the above two results:

$$\frac{\partial \ell(\boldsymbol{\theta} : \mathcal{D})}{\partial P(d^0 \mid c^0)} = 1.0382$$

Expectation Maximization (EM)

Tailored algorithm for optimizing likelihood function (Note: the red function is the convex likelihood function for the fully observed case that is used as a local approximate to the green function)



Intuition

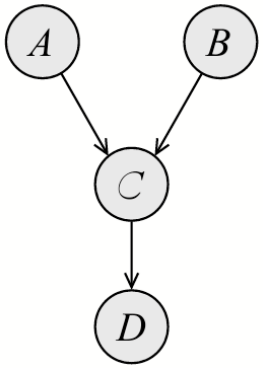
- Parameter estimation is easy given complete data
- Computing probability of missing data is "easy" (=inference) given parameters

Strategy

1. Pick a starting point for parameters
2. "Complete" the data using current parameters
3. Use the completed data as if it was real to do MLE estimate to produce new set of model parameters
4. Iterate steps 2 through 3
5. Guaranteed to improve at each iteration (ensures local maxima)

Expectation Maximization for BN (I)

Example: Consider the network shown and a partially specified data case $o = \langle a^1, ?, ?, d^0 \rangle$.



- Four possible completions (assume binary case): a^1, b^0, c^0, d^0
  a^1, b^0, c^1, d^0
  a^1, b^1, c^0, d^0
  a^1, b^1, c^1, d^0

Assume that our current $\boldsymbol{\theta}$ is:

$$
\begin{aligned}
\boldsymbol{\theta}_{a^1} &= 0.3 \\
\boldsymbol{\theta}_{b^1} &= 0.9 \\
\boldsymbol{\theta}_{c^1|a^0,b^0} &= 0.83 \\
\boldsymbol{\theta}_{c^1|a^0,b^1} &= 0.09 \\
\boldsymbol{\theta}_{c^1|a^1,b^0} &= 0.6 \\
\boldsymbol{\theta}_{c^1|a^1,b^1} &= 0.2 \\
\boldsymbol{\theta}_{d^1|c^0} &= 0.1 \\
\boldsymbol{\theta}_{d^1|c^1} &= 0.8.
\end{aligned}
$$

$$Q(B, C) = P(B, C \mid a^1, d^0, \boldsymbol{\theta})$$

$$
\begin{aligned}
Q(\langle b^1, c^1 \rangle) &= 0.3 \cdot 0.9 \cdot 0.2 \cdot 0.2 / 0.2196 = 0.0492 \\
Q(\langle b^1, c^0 \rangle) &= 0.3 \cdot 0.9 \cdot 0.8 \cdot 0.9 / 0.2196 = 0.8852 \\
Q(\langle b^0, c^1 \rangle) &= 0.3 \cdot 0.1 \cdot 0.6 \cdot 0.2 / 0.2196 = 0.0164 \\
Q(\langle b^0, c^0 \rangle) &= 0.3 \cdot 0.1 \cdot 0.4 \cdot 0.9 / 0.2196 = 0.0492,
\end{aligned}
$$

For each sample data m in data D, we will compute such a Q distribution over the possible completion of the sample (e.g. 4 cases as above in this example).

Expectation Maximization for BN (II)

- let H[m] denote the variables whose values are missing in the data instance o[m].
- Having the completed data cases, we can now do standard maximum likelihood estimation.
- We compute the *expected sufficient statistics*:

$$\bar{M}_{\boldsymbol{\theta}}[\boldsymbol{y}] = \sum_{m=1}^{M} \sum_{\boldsymbol{h}[m] \in Val(\boldsymbol{H}[m])} Q(\boldsymbol{h}[m]) \mathbb{1}\{\xi[m]\langle \boldsymbol{Y} \rangle = \boldsymbol{y}\}$$

- Use these expected sufficient statistics as if they were real in the MLE formula, for example:

$$\tilde{\boldsymbol{\theta}}_{d^1|c^0} = \frac{\bar{M}_{\boldsymbol{\theta}}[d^1, c^0]}{\bar{M}_{\boldsymbol{\theta}}[c^0]}$$

- In our previous example: suppose data consist of two instances o = ⟨a1,?,?,d0⟩ and o' = ⟨?, b1, ?, d1⟩. Then, using calculated Q and Q' :

$$\begin{aligned}
\bar{M}_{\boldsymbol{\theta}}[d^1, c^0] &= Q'(\langle a^1, c^0 \rangle) + Q'(\langle a^0, c^0 \rangle) \\
&= 0.1290 + 0.3423 = 0.4713 \\
\bar{M}_{\boldsymbol{\theta}}[c^0] &= Q(\langle b^1, c^0 \rangle) + Q(\langle b^0, c^0 \rangle) + Q'(\langle a^1, c^0 \rangle) + Q'(\langle a^0, c^0 \rangle) \\
&= 0.8852 + 0.0492 + 0.1290 + 0.3423 = 1.4057.
\end{aligned}$$

$$\Rightarrow \quad \tilde{\boldsymbol{\theta}}_{d^1|c^0} = \frac{0.4713}{1.4057} = 0.3353$$

Expectation Maximization for BN: General

- Observe that:

Very expensive if many missing variables in any given data sample.

$$\bar{M}_{\boldsymbol{\theta}}[\boldsymbol{y}] = \sum_{m=1}^{M} \sum_{\boldsymbol{h}[m] \in Val(\boldsymbol{H}[m])} Q(\boldsymbol{h}[m]) \mathbb{1}\{\xi[m]\langle \boldsymbol{Y} \rangle = \boldsymbol{y}\}$$

Significant computational ramification if viewed it as inference via Q

$= Q_m(y)$ It is inference using a distribution Q

- $Q_m(y)$ can be computed via belief propagation over clique tree, observation is evidence, and unobserved values are our variable, using current iteration parameters of BN. At calibration, we read out the marginals $Q_m(y)$. As such, we ran calibration/inference only once for each data sample (rather than once per each family $y=(X_i, Pa_i)$ in the network).

EM Strategy (rewritten 3 slides earlier)--Pick a starting point of parameters

1. Compute the expected sufficient statistics (E-Step)
2. Use the expected sufficient statistics in E-step to compute MLE estimate to produce new set of model parameters (M-Step)
3. Iterate (E-Step) through (M-Step)
4. Guaranteed to improve at each iteration (ensures local maxima)

Expectation Maximization for General BN (I)

- Initialize parameters to θ^0
- Iterate E-step and M-step
- In the t^{th} iteration:
  - Expectation (E-step):
    - Let o[m] be observed data in the m^{th} training instance.
    - For each m and each family $X_i, \mathbf{Pa}_i$,
      - compute $Qm(X_i, \mathbf{Pa}_i) = P(X_i, \mathbf{Pa}_i \mid o[m], \theta(t))$ (similar to gradient method)
    - Compute expected sufficient statistics for each values x, **u** on $X_i, \mathbf{Pa}_i$, respectivel

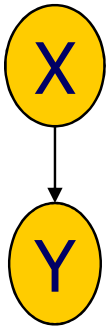
$$\overline{M}_{\theta^{(t)}}[X_i = x, \mathbf{Pa}_i = \mathbf{u}] = \sum_m P(X_i = x, \mathbf{Pa}_i = \mathbf{u} \mid o[m], \theta^{(t)})$$

  - Maximization (M-step):
  - Treat the expected sufficient statistics as observed and set the parameters to the MLE with respect to t

$$\theta^{(t+1)}_{X_i = x \mid \mathbf{Pa}_i = \mathbf{u}} = \frac{\overline{M}_{\theta^{(t)}}[X_i = x, \mathbf{Pa}_i = \mathbf{u}]}{\overline{M}_{\theta^{(t)}}[\mathbf{Pa}_i = \mathbf{u}]}$$

EM Summary

Initial network



+

Training data

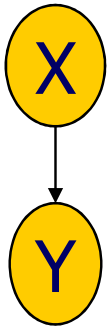
X	Y
?	y^0
x^0	y^1
?	y^0

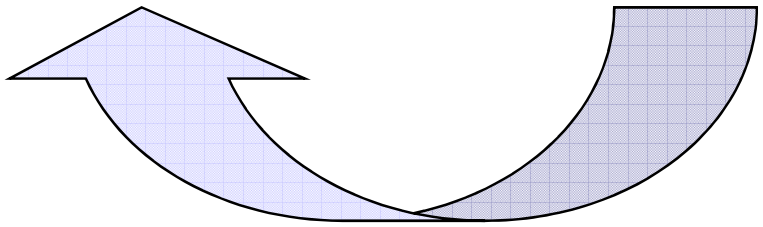
E-Step
(inference)

Expected counts
N(X)
N(X,Y)

M-Step
(reparameterize)

Updated network





Iterate

-Lee

EM Summary (II)

- Formal Guarantees:
- Each iteration improves the likelihood:
- $L(D:\Theta^{(t+1)}) \geq L(D:\Theta^{(t)})$
- If $\Theta^{(t+1)}=\Theta^{(t)}$, then $\Theta(t)$ is a stationary point of $L(D:\Theta)$
  - This often means a local maxima.

- Main cost:
- Computations of sufficient statistics in E-Step
  - Requires inference for each instance in training set
  - Computational cost exactly the same as in gradient ascent!

- Issues:
- Highly sensitive to starting parameters (random or guessing from another source)
- Avoiding bad local maxima (multiple restarts)