Readings: K&F 18.6, 19.1, 19.2

Learning with Partially Observed Data

Lecture 12 – May 4, 2011
CSE 515, Statistical Methods, Spring 2011

Instructor: Su-In Lee
University of Washington, Seattle

Model Selection

- So far, we focused on single model
  - Given D={**X**[1],…,**X**[M]}, find best scoring model $\tilde{G} = \arg\max_G P(G \mid D)$
  - Use it to predict next example $P(\mathbf{X}[M+1] \mid D, \tilde{G})$
- Implicit assumption
  - Making predictions based on the Bayesian estimation rule:

    $$P(\mathbf{X}[M+1] \mid D) = \sum_G P(\mathbf{X}[M+1] \mid D, G) P(G \mid D)$$
  - Best scoring model dominates the weighted sum

    $$P(\mathbf{X}[M+1] \mid D) \approx P(\mathbf{X}[M+1] \mid D, \tilde{G})$$
    - Valid with many data instances (very large M)
- Pros:
  - We get a single structure
  - Allows for efficient use in our prediction tasks
- Cons:
  - Committing to the independencies of a particular structure
  - Other structures with similar score might be probable given D

Model Selection

- Density estimation
  - Picking one structure may suffice if it distribution $P(\mathbf{X}[M+1] \mid D, G)$ is similar for different high-scoring structures.
- Structure discovery
  - Several networks with similar scores → one or several of them might be close to the "true" structure, but we cannot distinguish between them given the data D.
  - Drawing a conclusion about the structure from one of the networks can be wrong
  - Thus, instead of picking one of the high-scoring structures, we should focus on estimating the "confidence" of the structural properties we are interested in.
  - Define features f(G) (e.g., edge, sub-structure, d-sep property)
  - Compute $P(f \mid D) = \sum_G f(G) P(G \mid D)$
    - Requires summing over exponentially many structures
    - We can reduce the computation assuming a certain ordering

Model Averaging Given an Order

- Assumptions $X_1, X_2, \cdots, X_{i-1}, X_i, X_{i+1}, \cdots, X_{n-1}, X_n$
  - Known total order of variables α
  - Maximum in-degree for variables d
- Marginal likelihood

Using decomposability assumption on prior P(G|α)

$$\begin{aligned} P(D \mid \alpha) &= \sum_{G \in G_d, \alpha} P(D \mid G) P(G \mid \alpha) \\ &= \sum_{G \in G_d, \alpha} \prod_i \exp\{FamScore_B(X_i \mid Pa_{X_i}^G : D)\} \\ &= \prod_i \sum_{\mathbf{U} \in \{\mathbf{U}: \mathbf{U} < X_i \in \alpha, |\mathbf{U}| < d\}} \exp\{FamScore_B(X_i \mid U_i : D)\} \end{aligned}$$

$$\sum_g (f_1^g)(f_2^g) \cdots (f_n^g) = (f_1^{g1} + \cdots + f_1^{gm}) \cdots (f_n^{g1} + \cdots + f_n^{gm})$$

Since given ordering α, parent choices are independent

Cost per family: $O(n^d)$

Total cost: $O(n^{d+1})$

Model Averaging Given an Order

- Posterior probability of a general feature f

$$P(f \mid \alpha, D) = \frac{P(f, D \mid \alpha)}{P(D \mid \alpha)} = \frac{\sum_{G \in G_d, \alpha} f(G) P(D \mid G) P(G \mid \alpha)}{\prod_i \sum_{\mathbf{U} \in \{\mathbf{U} : \mathbf{U} < X_i \in \alpha, |\mathbf{U}| < d\}} \exp\{FamScore_B(X_i \mid U_i : D)\}}$$

- f: particular choice of parents **U** for X_i

$$P(Pa^G_{X_i} = \mathbf{U} \mid D, \alpha) = \frac{\exp\{FamScore_B(X_i \mid U : D)\}}{\sum_{\mathbf{U} \in \{\mathbf{U} : \mathbf{U} < X_i \in \alpha, |\mathbf{U}| < d\}} \exp\{FamScore_B(X_i \mid U_i : D)\}}$$

← All terms cancel out

- f: existence of a particular edge between $X_j \rightarrow X_i$

$$P(X_j \in Pa^G_{X_i} \mid D, \alpha) = \frac{\sum_{\mathbf{U} \in \{\mathbf{U} : X_j \in \mathbf{U} \text{ and } \mathbf{U} < X_i \in \alpha, |\mathbf{U}| < d\}} \exp\{FamScore_B(X_i \mid U_i : D)\}}{\sum_{\mathbf{U} \in \{\mathbf{U} : \mathbf{U} < X_i \in \alpha, |\mathbf{U}| < d\}} \exp\{FamScore_B(X_i \mid U_i : D)\}}$$

Model Averaging

- We cannot assume that order is known
- Solution: Sample from posterior distribution of P(G|D)
  - If we manage to sample graphs $G_1, .., G_K$ from P(G|D)
  - Estimate feature probability by $P(f \mid D) \approx \frac{1}{K} \sum_{i=1}^{K} f(G_d)$
  - Sampling can be done by MCMC (Markov chain Monte Carlo)
    - Next week

Notes on Learning Local Structures

- Beyond table CPDs

- Define score with local structures
  - Example: in tree CPDs, score decomposes by leaves (not by X_i and a particular value on Par X_i)

- Prior may need to be extended
  - Example: in tree CPDs, penalty for tree structure per CPD (depth of the tree)

- Extend search operators to local structure
  - Example: in tree CPDs, we need to search for tree structure
  - Can be done by local encapsulated search or by defining new global operations

Structure Search: Summary

- Discrete optimization problem

- In general, NP-Hard
  - Need to resort to heuristic search
  - In practice, search is relatively fast (~100 vars in ~10 min):
    - Decomposability
    - Sufficient statistics

- In some cases, we can reduce the search problem to an easy optimization problem
  - Example: learning trees, a fixed ordering α

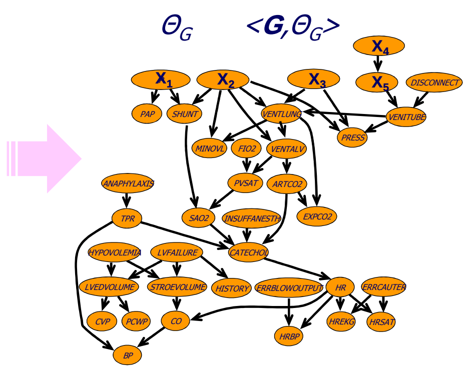
Let’s turn to the main topic for today…

LEARNING WITH PARTIALLY OBSERVED DATA

Training Data D

D *Training instance* ↓

X_1	3 1 0 -1 1 1 0 0 2 9 8 …
X_2	1 1 1 1 0 0 7 2 3 6 5 …
X_3	0 0 1 0 1 0 8 2 1 2 3 …
X_4	1 2 5 -2 3 0 1 3 4 5 …
:	:
X_{N-1}	1 3 2 3 6 …
X_N	0 7 4 -4 7 …



- Until now, we assumed that the training data is fully observed
  - Each instance assigns values to all the variables in our domain

Incomplete Data

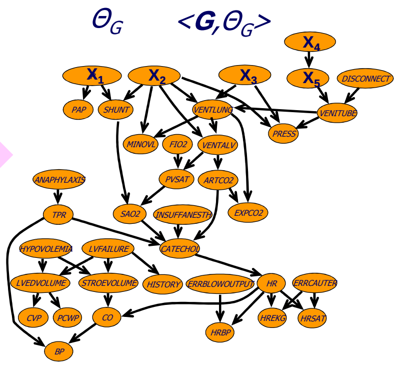
- In reality, this assumption might not be true.

Training instance ↓

D

X_1	3 1 **?** -1 1 1 **?** 0 2 9 8 …
X_2	1 **?** 1 1 0 0 7 2 3 6 5 …
X_3	0 0 1 **?** 1 0 8 2 **?** 2 3 …
X_4	1 2 5 -2 **?** 0 1 3 4 5 …
X_{N-1}	**? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ?**
X_N	0 7 4 **?** 7

Lung cancer?



- Missing values, Hidden variables

- Challenges
  - Foundational – is the learning task well defined?
  - Computational – how can we learn with missing data?

11

Treating Missing Data

- How should we treat missing data?
  - Based on data missing mechanism

- Case I: A coin is tossed on a table, occasionally it drops and measurements are not taken (random missing)
  - Sample sequence: H,T,?,?,T,?,H
  - Treat missing data by ignoring it

- Case II: A coin is tossed, but only heads are reported (deliberate missing values)
  - Sample sequence: H,?,?,?,H,?,H
  - Treat missing data by filling it with Tails

⇓

We need to consider the data missing mechanism

12

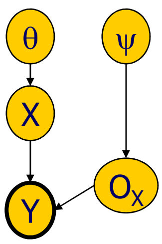
Modeling Data Missing Mechanism

- Let's try to model the data missing mechanism
- $X = \{X_1, \ldots, X_n\}$ are random variables
- $O_X = \{O_{X_1}, \ldots, O_{X_n}\}$ are *observability variables*
  - Always observed
- $Y = \{Y_1, \ldots, Y_n\}$ new random variables
  - $Val(Y_i) = Val(X_i) \cup \{?\}$
  - Y_i is a deterministic function of X_i and O_{X_1}:

$$Y_i = \begin{cases} X_i & O_{X_i} = o^1 \\ ? & O_{X_i} = o^0 \end{cases}$$

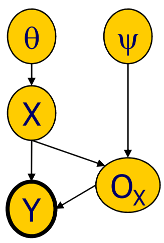
Modeling Missing Data Mechanism

Case I (random missing values)



$$P(Y=H) = \theta\psi$$
$$P(Y=T) = (1-\theta)\psi$$
$$P(Y=?) = (1-\psi)$$
$$L(D:\theta,\psi) = \theta^{M_H} \cdot (1-\theta)^{M_T} \cdot \psi^{M_H+M_T} \cdot (1-\psi)^{M_?}$$

Case II (deliberate missing values)



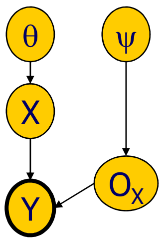
MLE

$$\hat{\theta} = \frac{M_H}{M_H + M_T}$$

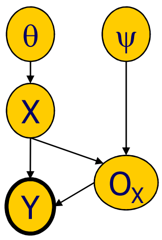
$$\hat{\psi} = \frac{M_H + M_T}{M_H + M_T + M_?}$$

Modeling Missing Data Mechanism

Case I
(random missing values)



Case II
(deliberate missing values)



MLE ?

$\hat{\theta} =$ **?**

$\hat{\psi} =$

$$P(Y=H) = \theta\psi_{O_X|H}$$
$$P(Y=T) = (1-\theta)\psi_{O_X|T}$$
$$P(Y=?) = \theta(1-\psi_{O_X|H}) + (1-\theta)(1-\psi_{O_X|T})$$

$$L(D:\theta,\psi) = \theta^{M_H} \cdot (1-\theta)^{M_T} \cdot \psi_{O_X|H}^{M_H} \cdot \psi_{O_X|T}^{M_T} \cdot \left(\theta(1-\psi_{O_X|H}) + (1-\theta(1-\psi_{O_X|T})\right)^{M_?}$$

Decoupling of Observation Mechanism

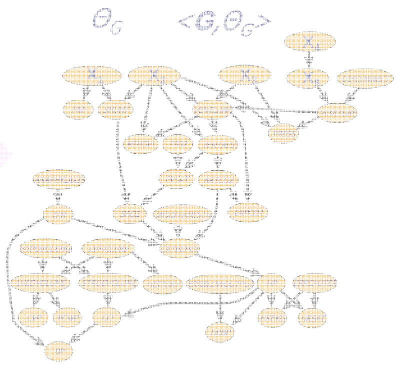
- When can we ignore the missing data mechanism and focus only on the likelihood?
  - Missing Completely at Random (MCAR)
    - For every X_i, Ind(X_i;O_{X_i}), a very strong assumption
    - Sufficient but not necessary for the decomposition of the likelihood
  - Missing at Random (MAR) is sufficient
    - The probability that the value of X_i is missing is independent of its actual value, given other observed values
- In both cases, the likelihood decomposes
  - When there are missing values in D, try to model such that MAR holds.

Incomplete Data

- In reality, this assumption might not be true.

D

X_1	3 1 ? -1 1 1 ? 0 2 9 8 …
X_2	1 ? 1 1 0 0 7 2 3 6 5 …
X_3	0 0 1 ? 1 0 8 2 ? 2 3 …
X_4	1 2 5 -2 ? 0 1 3 4 5 …
Lung cancer?	
X_{N-1}	? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ?
X_N	0 7 4 ? 7



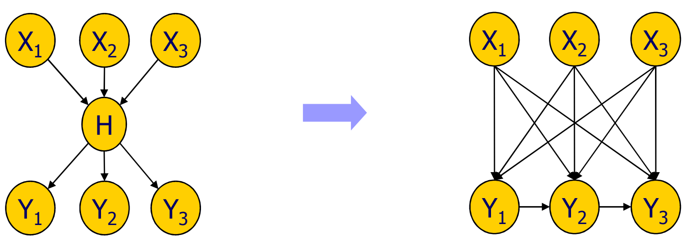
- Missing values, **Hidden variables**

- Challenges
  - Foundational – is the learning task well defined?
  - Computational – how can we learn with missing data?

Hidden (Latent) Variables

- Attempt to learn a model with hidden variables
  - In this case, MCAR always holds (variable is always missing)

- Why should we care about unobserved variables?



17 parameters

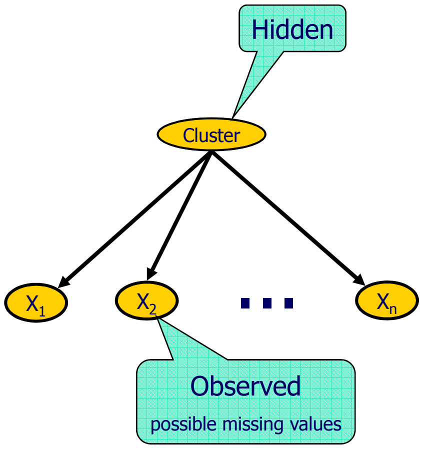
59 parameters

Hidden (Latent) Variables

- Hidden variables also appear in **clustering**
- Naïve Bayes model:
  - Class variable is hidden
  - Observed attributes are independent given the class

D

X_1	3 1 0 -1 1 1 0 0 2 9 8 …
X_2	1 1 1 1 0 0 7 2 3 6 5 …
X_3	0 0 1 0 1 0 8 2 1 2 3 …
X_4	1 2 5 -2 3 0 1 3 4 5 …
:	:
X_{N-1}	1 3 2 3 6 …
X_N	0 7 4 -4 7 …
H	1 2 1 1 3 3 1 1 2 2 1 1 …



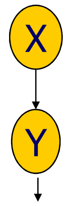
How do missing data affect the likelihood function?

Likelihood for Complete Data

Input Data:

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

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

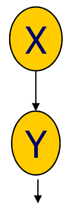
- Likelihood decomposes by variables
- Likelihood decomposes within CPDs
- Likelihood function is log-concave → unique global maximum that has a simple analytic closed form.

Likelihood for Incomplete Data

Input Data:

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	$\theta_{y0\|x1}$	$\theta_{y1\|x1}$

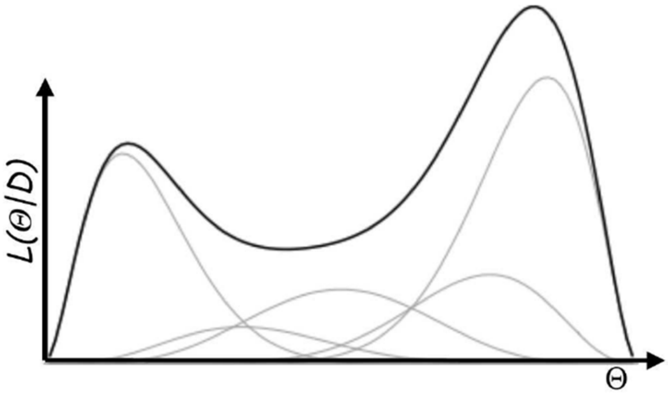
Likelihood:

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

- Likelihood does not decompose by variables
- Likelihood does not decompose within CPDs
- Computing likelihood per instance requires inference!

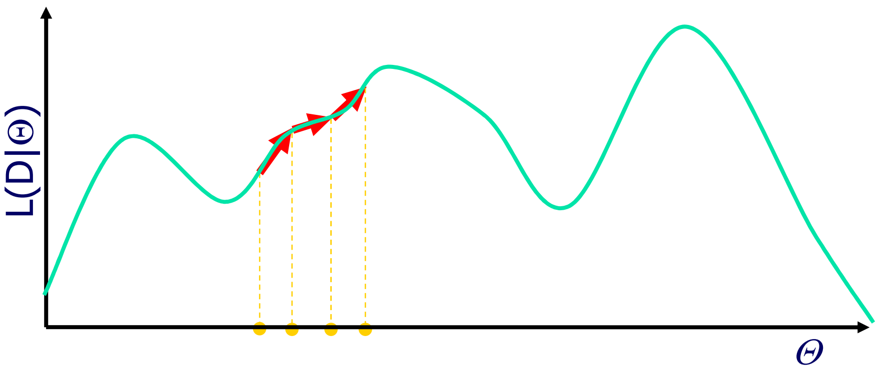
Likelihood with Missing Data

- Multimodal likelihood function with incomplete data
  - Likelihood function is not log-concave → local maxima cannot be obtained by a simple analytic closed form



MLE from Incomplete Data

- Take steps proportional to the positive of the gradient.

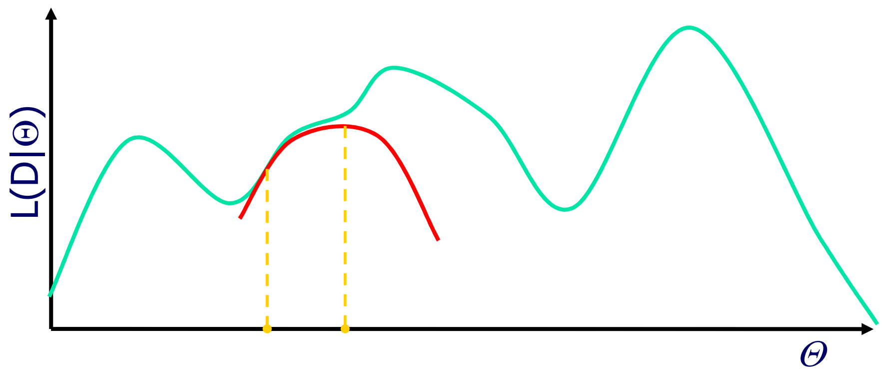


Gradient Ascent:

- Follow gradient of likelihood w.r.t. to parameters
- Add line search and conjugate gradient methods to get fast convergence

MLE from Incomplete Data

- Nonlinear optimization problem

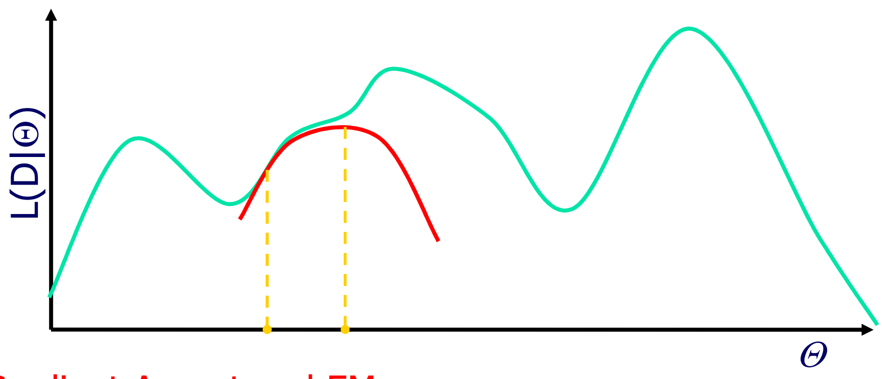


Expectation Maximization (EM):
- Use "current point" to construct alternative function (which is "nice")
- Guaranty: maximum of new function has better score than current point

25

MLE from Incomplete Data

- Nonlinear optimization problem

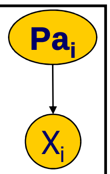


Gradient Ascent and EM
- Find local maxima
- Require multiple restarts to find approx. to the global maximum
- Require computations in each iteration

26

Gradient Ascent

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



- Theorem:

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

each training instance

observed data in the m-th instance

- Proof:

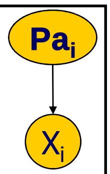
$$\frac{\partial \log P(D|\Theta)}{\partial \theta_{x_i,pa_i}} = \sum_m \frac{\partial \log P(o[m]|\Theta)}{\partial \theta_{x_i,pa_i}}$$

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

How do we compute ? $\dfrac{\partial P(o[m]|\Theta)}{\partial \theta_{x_i,pa_i}}$

27

Gradient Ascent



$$\frac{\partial P(o[m]|\Theta)}{\partial \theta_{x_i,pa_i}} = \sum_{\mathbf{X}:\mathbf{X}<\mathbf{O}>=o[m]} \frac{\partial P(\mathbf{X}|\Theta)}{\partial \theta_{x_i,pa_i}}$$

$$= \sum_{\mathbf{X}:\mathbf{X}<\mathbf{O}>=o[m],\mathbf{X}<X_i,Pa_i>=<x_i,pa_i>} \frac{P(\mathbf{X}|\Theta)}{\theta_{x_i,pa_i}}$$

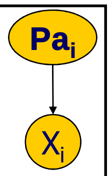
$$= \frac{1}{\theta_{x_i,pa_i}} P(x_i, pa_i, o[m]|\Theta)$$

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

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

28

Gradient Ascent



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

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

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

- Requires computation: $P(x_i,pa_i|o[m],\Theta)$ for all X_i, m
- Can be done with clique-tree algorithm, since X_i,Pa_i are in the same clique

29

Gradient Ascent Summary

- Pros
  - Flexible, can be extended to non table CPDs
- Cons
  - Need to project gradient onto space of legal parameters
  - For reasonable convergence, need to combine with advanced methods (conjugate gradient, line search)

30

Expectation Maximization (EM)

- Tailored algorithm for optimizing likelihood functions
- Intuition
  - Parameter estimation is easy given complete data
  - Computing probability of missing data is "easy" (=inference) given parameters
- Strategy
  - Pick a starting point for parameters
  - "Complete" the data using current parameters
  - Estimate parameters relative to data completion
  - Iterate
  - Procedure guaranteed to improve at each iteration

Expectation Maximization (EM)

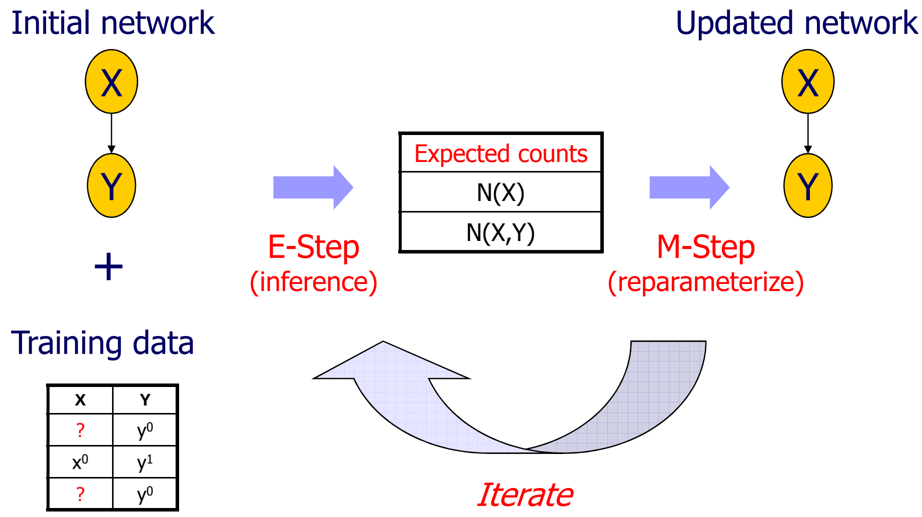
- Initialize parameters to θ^0
- Iterate E-step and M-step
- In the t-th iteration, we do
- Expectation (E-step):
  - Let o[m] be the observed data in the m-th training instance.
  - For each m and each family X_i,$\mathbf{Pa}_i$, compute P(X_i,$\mathbf{Pa}_i$ | o[m], $\theta^{(t)}$)
  - Compute the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

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

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

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

Expectation Maximization (EM)



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

33

Expectation Maximization (EM)

- Formal Guarantees:
  - $L(D:\Theta^{(t+1)}) \geq L(D:\Theta^{(t)})$
    - Each iteration improves the likelihood
  - If $\Theta^{(t+1)}=\Theta^{(t)}$, then $\Theta^{(t)}$ is a stationary point of $L(D:\Theta)$
    - Usually, this means a local maximum
- Main cost:
  - Computations of expected counts in E-Step
  - Requires inference for each instance in training set
    - Exactly the same as in gradient ascent!
- Reading material on EM
  - Please read Andrew Ng's lecture note

34

EM – Practical Considerations

- Initial parameters
  - Highly sensitive to starting parameters
  - Choose randomly
  - Choose by guessing from another source
- Stopping criteria
  - Small change in data likelihood
  - Small change in parameters
- Avoiding bad local maxima
  - Multiple restarts
  - Early pruning of unpromising starting points

Acknowledgement

- These lecture notes were generated based on the slides from Prof Eran Segal.