



Machine Learning 10-601

Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University

March 4, 2015

Today:

- Graphical models
- Bayes Nets:
 - EM
 - Mixture of Gaussian clustering
 - Learning Bayes Net structure (Chow-Liu)

Readings:

- Bishop chapter 8
- Mitchell chapter 6

Learning of Bayes Nets

- Four categories of learning problems
 - Graph structure may be known/unknown
 - Variable values may be fully observed / partly unobserved
- Easy case: learn parameters for graph structure is *known*, and data is *fully observed*
- Interesting case: graph *known*, data *partly known*
- Gruesome case: graph structure *unknown*, data *partly unobserved*

Learning CPTs from Fully Observed Data

- Example: Consider learning the parameter

$$\theta_{s|ij} \equiv P(S = 1 | F = i, A = j)$$

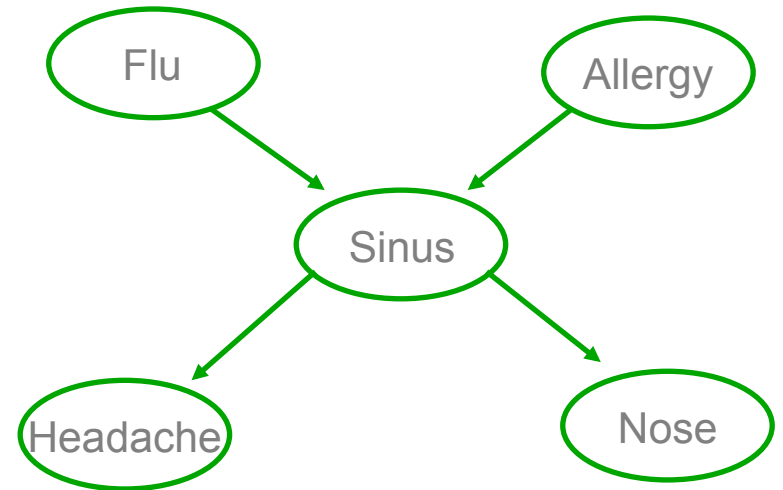
- Max Likelihood Estimate is

$$\theta_{s|ij} = \frac{\sum_{k=1}^K \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^K \delta(f_k = i, a_k = j)}$$

k^{th} training example

$\delta(x) = 1$ if $x=\text{true}$,
= 0 if $x=\text{false}$

- Remember why?



let's use $p(a,b)$ as shorthand for $p(A=a, B=b)$

MLE estimate of $\theta_{s|ij}$ from fully observed data

- Maximum likelihood estimate

$$\theta \leftarrow \arg \max_{\theta} \log P(\text{data}|\theta)$$

- Our case:

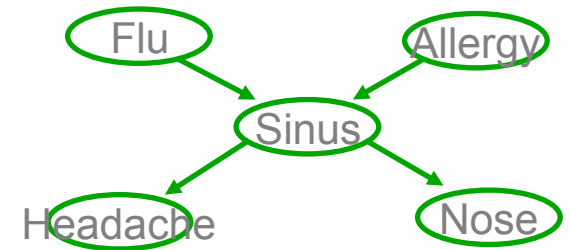
$$P(\text{data}|\theta) = \prod_{k=1}^K P(f_k, a_k, s_k, h_k, n_k)$$

$$P(\text{data}|\theta) = \prod_{k=1}^K P(f_k)P(a_k)P(s_k|f_k a_k)P(h_k|s_k)P(n_k|s_k)$$

$$\log P(\text{data}|\theta) = \sum_{k=1}^K \log P(f_k) + \log P(a_k) + \log P(s_k|f_k a_k) + \log P(h_k|s_k) + \log P(n_k|s_k)$$

$$\frac{\partial \log P(\text{data}|\theta)}{\partial \theta_{s|ij}} = \sum_{k=1}^K \frac{\partial \log P(s_k|f_k a_k)}{\partial \theta_{s|ij}}$$

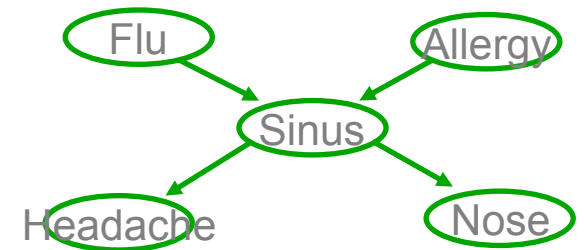
$$\theta_{s|ij} = \frac{\sum_{k=1}^K \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^K \delta(f_k = i, a_k = j)}$$



Estimate θ from partly observed data

- What if FAHN observed, but not S?
- Can't calculate MLE

$$\theta \leftarrow \arg \max_{\theta} \log \prod_k P(f_k, a_k, s_k, h_k, n_k | \theta)$$



- Let X be all *observed* variable values (over all examples)
- Let Z be all *unobserved* variable values
- Can't calculate MLE:

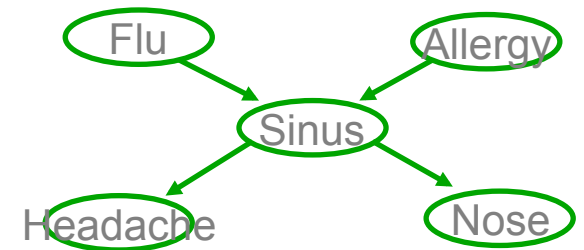
$$\theta \leftarrow \arg \max_{\theta} \log P(X, Z | \theta)$$

- WHAT TO DO?

Estimate θ from partly observed data

- What if FAHN observed, but not S?
- Can't calculate MLE

$$\theta \leftarrow \arg \max_{\theta} \log \prod_k P(f_k, a_k, s_k, h_k, n_k | \theta)$$



- Let X be all *observed* variable values (over all examples)
- Let Z be all *unobserved* variable values
- Can't calculate MLE:

$$\theta \leftarrow \arg \max_{\theta} \log P(X, Z | \theta)$$

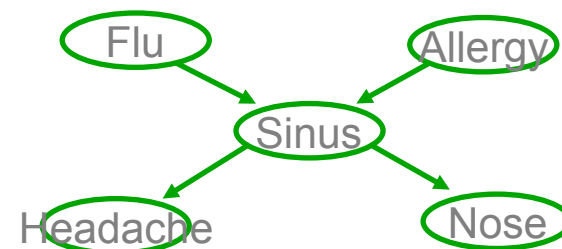
- EM seeks* to estimate:

$$\theta \leftarrow \arg \max_{\theta} E_{Z|X, \theta} [\log P(X, Z | \theta)]$$

* EM guaranteed to find local maximum

- EM seeks estimate:

$$\theta \leftarrow \arg \max_{\theta} E_{Z|X,\theta}[\log P(X, Z|\theta)]$$



- here, observed $X=\{F,A,H,N\}$, unobserved $Z=\{S\}$

$$\log P(X, Z|\theta) = \sum_{k=1}^K \log P(f_k) + \log P(a_k) + \log P(s_k|f_k a_k) + \log P(h_k|s_k) + \log P(n_k|s_k)$$

$$E_{P(Z|X,\theta)} \log P(X, Z|\theta) = \sum_{k=1}^K \sum_{i=0}^1 P(s_k = i | f_k, a_k, h_k, n_k) \\ [\log P(f_k) + \log P(a_k) + \log P(s_k | f_k a_k) + \log P(h_k | s_k) + \log P(n_k | s_k)]$$

EM Algorithm - Informally

EM is a general procedure for learning from partly observed data

Given observed variables X , unobserved Z ($X=\{F,A,H,N\}$, $Z=\{S\}$)

Begin with arbitrary choice for parameters θ

Iterate until convergence:

- E Step: estimate the values of unobserved Z , using θ
- M Step: use observed values plus E-step estimates to derive a better θ

Guaranteed to find local maximum.

Each iteration increases $E_{P(Z|X,\theta)}[\log P(X, Z|\theta')]$

EM Algorithm - Precisely

EM is a general procedure for learning from partly observed data

Given observed variables X , unobserved Z ($X=\{F,A,H,N\}$, $Z=\{S\}$)✓

Define $Q(\theta'|\theta) = E_{P(Z|X,\theta)}[\log P(X, Z|\theta')]$

Iterate until convergence:

- E Step: Use X and current θ to calculate $P(Z|X,\theta)$
- M Step: Replace current θ by

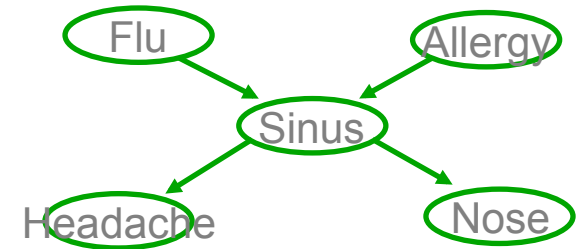
$$\theta \leftarrow \arg \max_{\theta'} Q(\theta'|\theta)$$

Guaranteed to find local maximum.

Each iteration increases $E_{P(Z|X,\theta)}[\log P(X, Z|\theta')]$

E Step: Use X, θ , to Calculate $P(Z|X,\theta)$

observed $X=\{F,A,H,N\}$,
unobserved $Z=\{S\}$



- How? Bayes net inference problem.

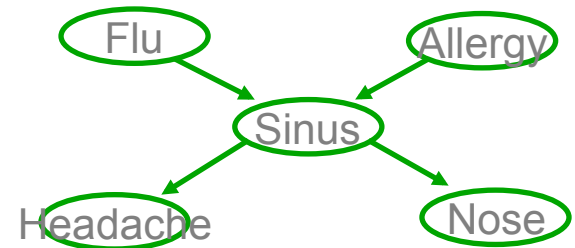
$$P(S_k = 1 | f_k a_k h_k n_k, \theta) =$$

$$P(S_k = 1 | f_k a_k h_k n_k, \theta) = \frac{P(S_k = 1, f_k a_k h_k n_k | \theta)}{P(S_k = 1, f_k a_k h_k n_k | \theta) + P(S_k = 0, f_k a_k h_k n_k | \theta)}$$

let's use $p(a,b)$ as shorthand for $p(A=a, B=b)$

EM and estimating $\theta_{s|ij}$

observed $X = \{F, A, H, N\}$, unobserved $Z = \{S\}$



E step: Calculate $P(Z_k|X_k; \theta)$ for each training example, k

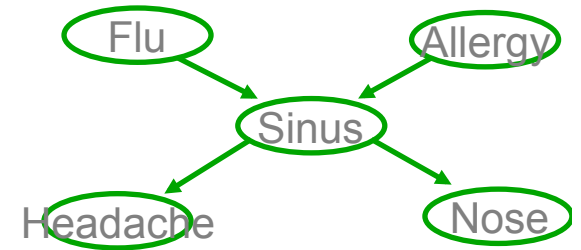
$$P(S_k = 1 | f_k a_k h_k n_k, \theta) = \underbrace{E[s_k]}_{P(Z_k|X_k; \theta)} = \frac{P(S_k = 1, f_k a_k h_k n_k | \theta)}{P(S_k = 1, f_k a_k h_k n_k | \theta) + P(S_k = 0, f_k a_k h_k n_k | \theta)}$$

M step: update all relevant parameters. For example:

$$\theta_{s|ij} \leftarrow \frac{\sum_{k=1}^K \delta(f_k = i, a_k = j) E[s_k]}{\sum_{k=1}^K \delta(f_k = i, a_k = j)}$$

Recall MLE was: $\theta_{s|ij} = \frac{\sum_{k=1}^K \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^K \delta(f_k = i, a_k = j)}$

EM and estimating θ



More generally,

Given observed set X , unobserved set Z of boolean values

E step: Calculate for each training example, k

the expected value of each unobserved variable in each training example

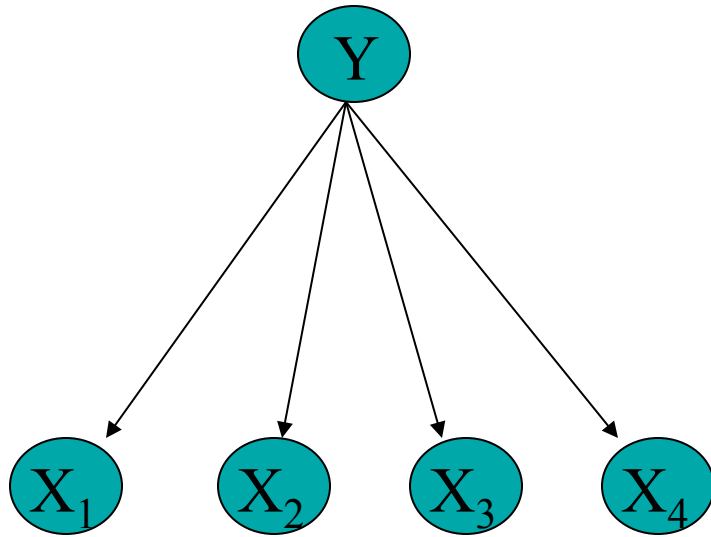
M step:

Calculate θ similar to MLE estimates, but replacing each count by its expected count

$$\delta(Y = 1) \rightarrow E_{Z|X,\theta}[Y] \qquad \delta(Y = 0) \rightarrow (1 - E_{Z|X,\theta}[Y])$$

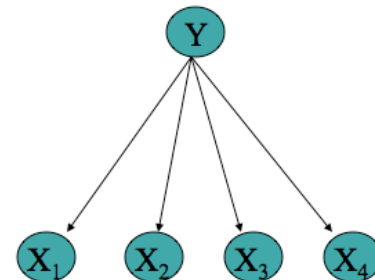
Using Unlabeled Data to Help Train Naïve Bayes Classifier

Learn $P(Y|X)$



Y	X1	X2	X3	X4
1	0	0	1	1
0	0	1	0	0
0	0	0	1	0
?	0	1	1	0
?	0	1	0	1

EM and estimating θ



Given observed set X , unobserved set Y of boolean values

E step: Calculate for each training example, k

the expected value of each unobserved variable Y

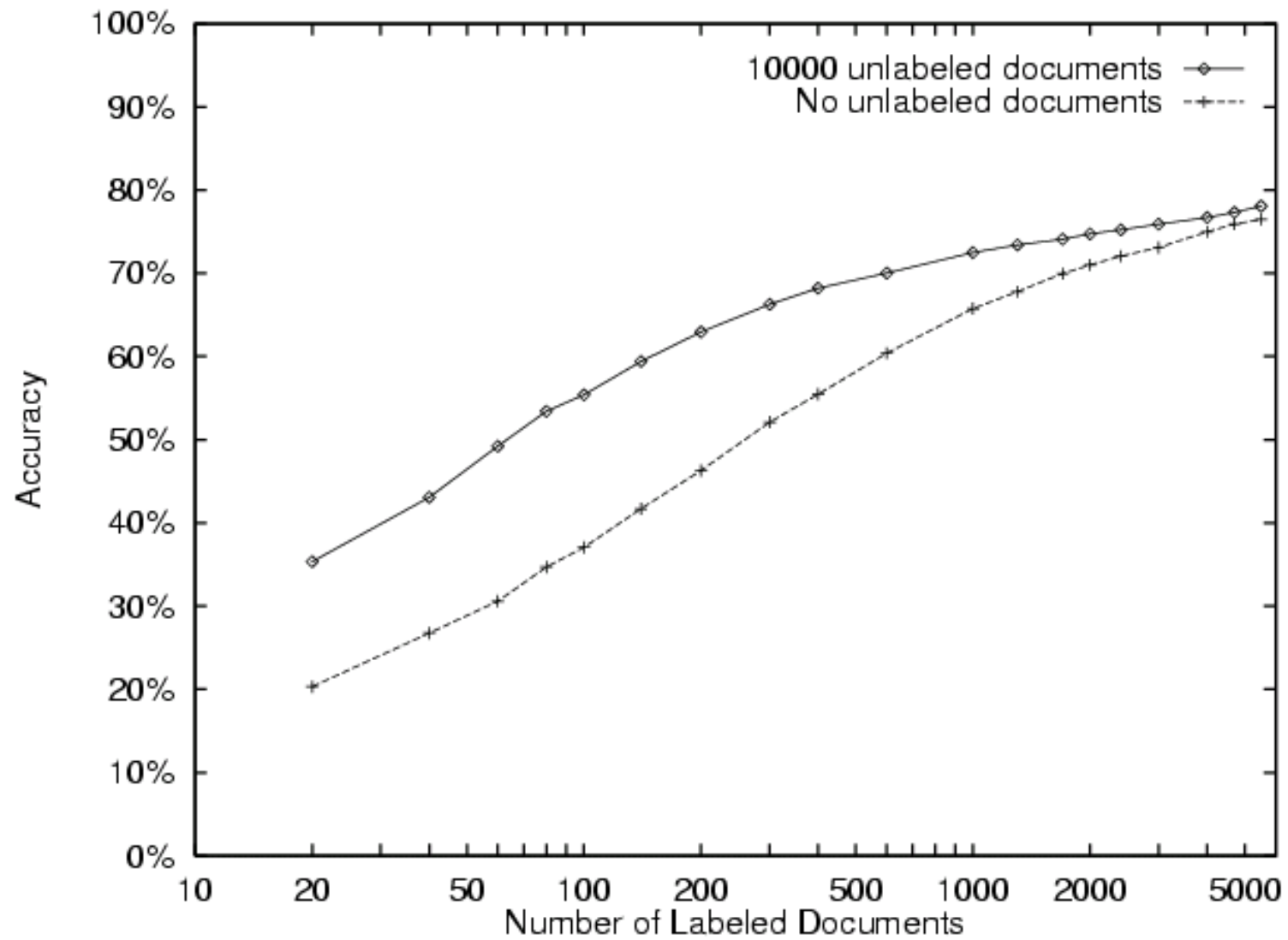
$$E_{P(Y|X_1 \dots X_N)}[y(k)] = P(y(k) = 1 | x_1(k), \dots, x_N(k); \theta) = \frac{P(y(k) = 1) \prod_i P(x_i(k) | y(k) = 1)}{\sum_{j=0}^1 P(y(k) = j) \prod_i P(x_i(k) | y(k) = j)}$$

M step: Calculate estimates similar to MLE, but
replacing each count by its expected count

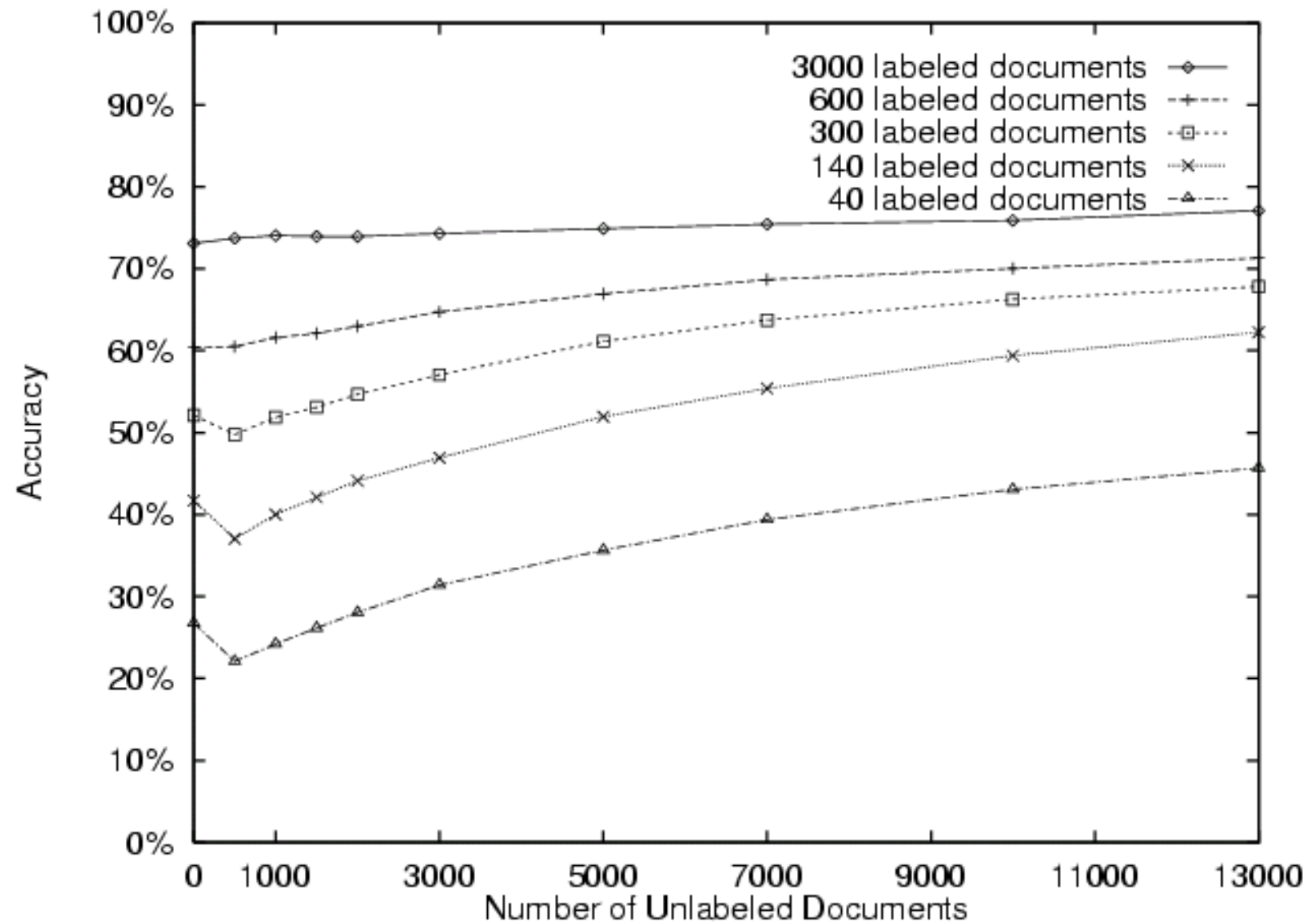
$$\theta_{ij|m} = \hat{P}(X_i = j | Y = m) = \frac{\sum_k P(y(k) = m | x_1(k) \dots x_N(k)) \delta(x_i(k) = j)}{\sum_k P(y(k) = m | x_1(k) \dots x_N(k))}$$

$$\text{MLE would be: } \hat{P}(X_i = j | Y = m) = \frac{\sum_k \delta((y(k) = m) \wedge (x_i(k) = j))}{\sum_k \delta(y(k) = m)}$$

20 Newsgroups



20 Newsgroups



Unsupervised clustering

Just extreme case for EM with
zero labeled examples...

Clustering

- Given set of data points, group them
- Unsupervised learning
- Which patients are similar? (or which earthquakes, customers, faces, web pages, ...)

Mixture Distributions

Model joint $P(X_1 \dots X_n)$ as mixture of multiple distributions.

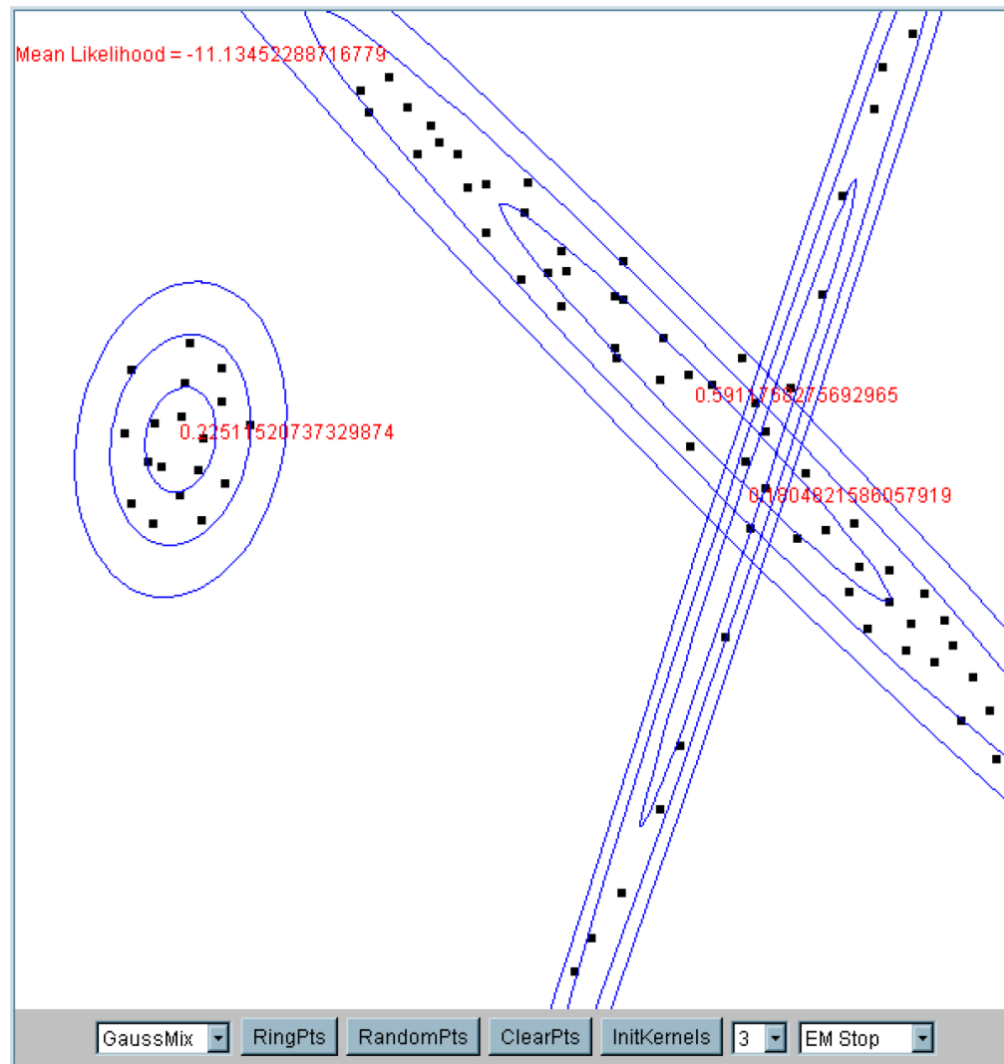
Use discrete-valued random var Z to indicate which distribution is being use for each random draw

So
$$P(X_1 \dots X_n) = \sum_i P(Z = i) P(X_1 \dots X_n | Z)$$

Mixture of *Gaussians*:

- Assume each data point $X = \langle X_1, \dots, X_n \rangle$ is generated by one of several Gaussians, as follows:
 1. randomly choose Gaussian i , according to $P(Z=i)$
 2. randomly generate a data point $\langle x_1, x_2 \dots x_n \rangle$ according to $N(\mu_i, \Sigma_i)$

Mixture of Gaussians



EM for Mixture of Gaussian Clustering

Let's simplify to make this easier:

1. assume $X = \langle X_1 \dots X_n \rangle$, and the X_i are conditionally independent given Z .

$$P(X|Z = j) = \prod_i N(X_i|\mu_{ji}, \sigma_{ji})$$

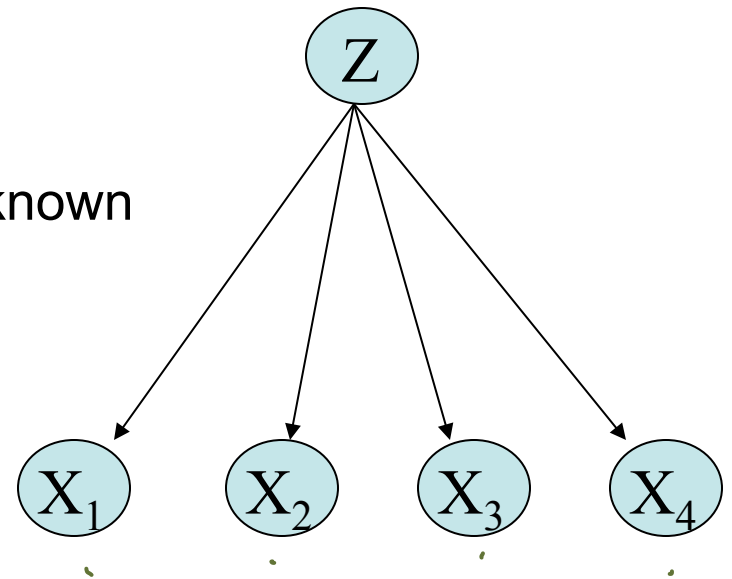
2. assume only 2 clusters (values of Z), and $\forall i, j, \sigma_{ji} = \sigma$

$$P(X) = \sum_{j=1}^2 P(Z = j|\pi) \prod_i N(x_i|\mu_{ji}, \sigma)$$

3. Assume σ known, $\pi_1 \dots \pi_K, \mu_{1i} \dots \mu_{Ki}$ unknown

Observed: $X = \langle X_1 \dots X_n \rangle$

Unobserved: Z

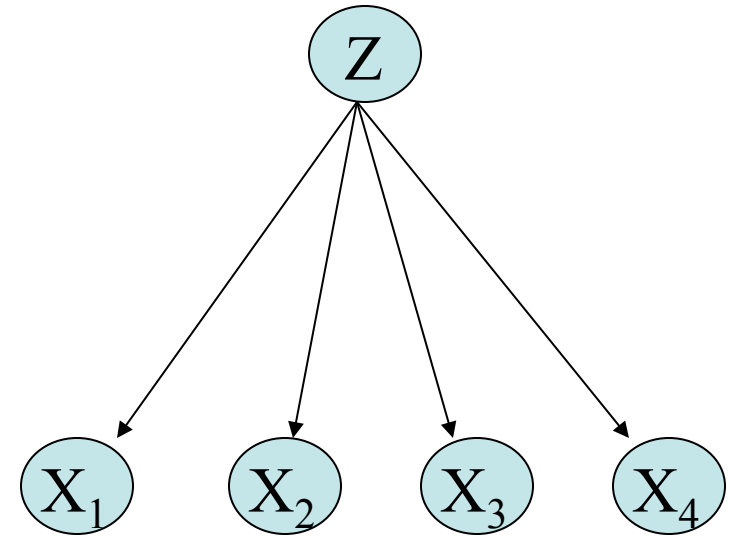


EM

Given observed variables X , unobserved Z

Define $Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X, Z|\theta')]$

where $\theta = \langle \pi, \mu_{ji} \rangle$



Iterate until convergence:

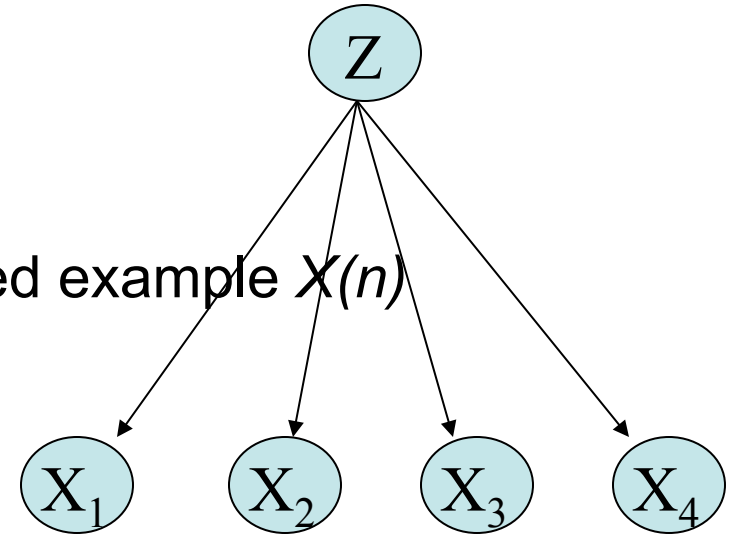
- E Step: Calculate $P(Z(n)|X(n), \theta)$ for each example $X(n)$. Use this to construct $Q(\theta'|\theta)$

- M Step: Replace current θ by
$$\theta \leftarrow \arg \max_{\theta'} Q(\theta'|\theta)$$

EM – E Step

Calculate $P(Z(n)|X(n), \theta)$ for each observed example $X(n)$

$X(n) = \langle x_1(n), x_2(n), \dots, x_T(n) \rangle$.



$$P(z(n) = k | x(n), \theta) = \frac{P(x(n) | z(n) = k, \theta) P(z(n) = k | \theta)}{\sum_{j=0}^1 P(x(n) | z(n) = j, \theta) P(z(n) = j | \theta)}$$

$$P(z(n) = k | x(n), \theta) = \frac{\prod_i P(x_i(n) | z(n) = k, \theta) P(z(n) = k | \theta)}{\sum_{j=0}^1 \prod_i P(x_i(n) | z(n) = j, \theta) P(z(n) = j | \theta)}$$

$$P(z(n) = k | x(n), \theta) = \frac{\prod_i N(x_i(n) | \mu_{k,i}, \sigma) (\pi^k (1 - \pi)^{(1-k)})}{\sum_{j=0}^1 [\prod_i N(x_i(n) | \mu_{j,i}, \sigma) (\pi^j (1 - \pi)^{(1-j)})]}$$

EM – M Step

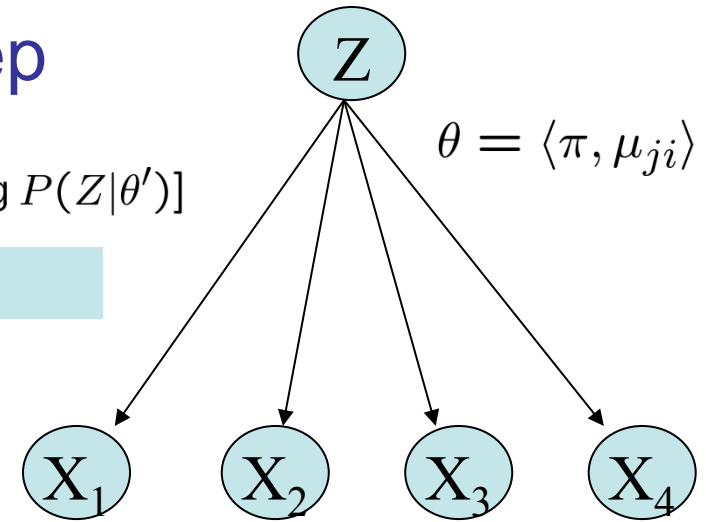
First consider update for π

$$Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X, Z|\theta')] = E[\log P(X|Z, \theta') + \log P(Z|\theta')]$$

π' has no influence

$$\pi \leftarrow \arg \max_{\pi'} E_{Z|X,\theta}[\log P(Z|\pi')]$$

$z=1$ for nth example



$$E_{Z|X,\theta}[\log P(Z|\pi')] = E_{Z|X,\theta}[\log (\pi' \sum_n z(n) (1 - \pi')^{\sum_n (1-z(n))})]$$

$$= E_{Z|X,\theta} \left[\left(\sum_n z(n) \right) \log \pi' + \left(\sum_n (1 - z(n)) \right) \log(1 - \pi') \right]$$

$$= \left(\sum_n E_{Z|X,\theta}[z(n)] \right) \log \pi' + \left(\sum_n E_{Z|X,\theta}[(1 - z(n))] \right) \log(1 - \pi')$$

$$\frac{\partial E_{Z|X,\theta}[\log P(Z|\pi')]}{\partial \pi'} = \left(\sum_n E_{Z|X,\theta}[z(n)] \right) \frac{1}{\pi'} + \left(\sum_n E_{Z|X,\theta}[(1 - z(n))] \right) \frac{(-1)}{1 - \pi'}$$

$$\pi \leftarrow \frac{\sum_{n=1}^N E[z(n)]}{\left(\sum_{n=1}^N E[z(n)] \right) + \left(\sum_{n=1}^N (1 - E[z(n)]) \right)} = \frac{1}{N} \sum_{n=1}^N E[z(n)]$$

EM – M Step

Now consider update for μ_{ji}

$$Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X, Z|\theta')] = E[\log P(X|Z, \theta') + \log P(Z|\theta')]$$

μ_{ji}' has no influence

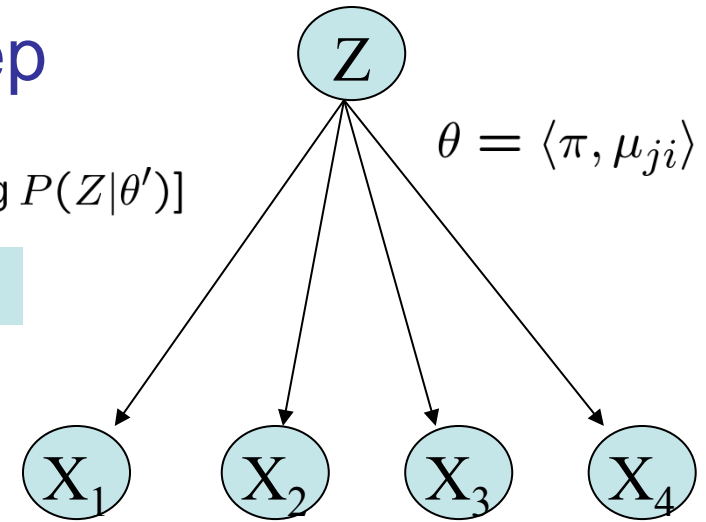
$$\mu_{ji} \leftarrow \arg \max_{\mu_{ji}'} E_{Z|X,\theta}[\log P(X|Z, \theta')]$$

...

$$\mu_{ji} \leftarrow \frac{\sum_{n=1}^N P(z(n) = j | x(n), \theta) x_i(n)}{\sum_{n=1}^N P(z(n) = j | x(n), \theta)}$$

Compare above to
MLE if Z were
observable:

$$\mu_{ji} \leftarrow \frac{\sum_{n=1}^N \delta(z(n) = j) x_i(n)}{\sum_{n=1}^N \delta(z(n) = j)}$$

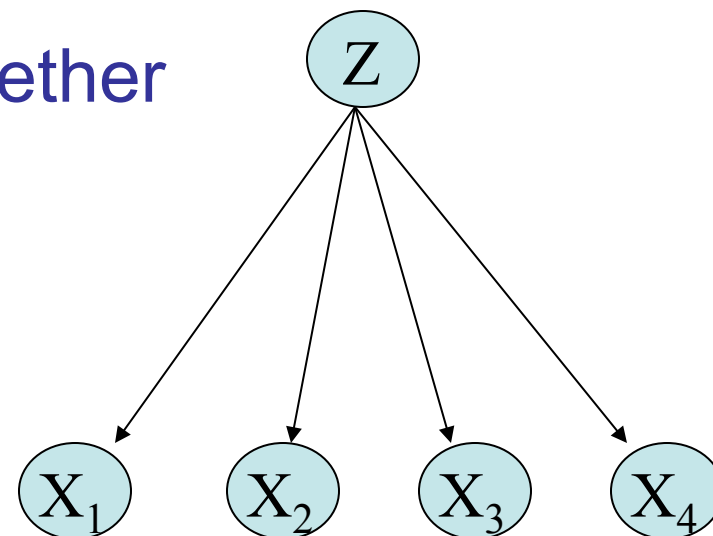


EM – putting it together

Given observed variables X , unobserved Z

Define $Q(\theta'|\theta) = E_{Z|X,\theta}[\log P(X, Z|\theta')]$

where $\theta = \langle \pi, \mu_{ji} \rangle$



Iterate until convergence:

- E Step: For each observed example $X(n)$, calculate $P(Z(n)|X(n), \theta)$

$$P(z(n) = k | x(n), \theta) = \frac{[\prod_i N(x_i(n) | \mu_{k,i}, \sigma)] (\pi^k (1 - \pi)^{(1-k)})}{\sum_{j=0}^1 [\prod_i N(x_i(n) | \mu_{j,i}, \sigma)] (\pi^j (1 - \pi)^{(1-j)})}$$

- M Step: Update $\theta \leftarrow \arg \max_{\theta'} Q(\theta'|\theta)$

$\pi \leftarrow \frac{1}{N} \sum_{n=1}^N E[z(n)]$

(Handwritten green note: $p(z=1)$ with an arrow pointing to π)

$$\mu_{ji} \leftarrow \frac{\sum_{n=1}^N P(z(n) = j | x(n), \theta) x_i(n)}{\sum_{n=1}^N P(z(n) = j | x(n), \theta)}$$

What you should know about EM

- For learning from partly unobserved data
- MLE of $\theta = \arg \max_{\theta} \log P(\text{data}|\theta)$
- EM estimate: $\theta = \arg \max_{\theta} E_{Z|X,\theta}[\log P(X, Z|\theta)]$
Where X is observed part of data, Z is unobserved
- Nice case is Bayes net of boolean vars:
 - M step is like MLE, with with unobserved values replaced by their expected values, given the other observed values
- EM for training Bayes networks
- Can also develop MAP version of EM
- Can also derive your own EM algorithm for your own problem
 - write out expression for $E_{Z|X,\theta}[\log P(X, Z|\theta)]$
 - E step: for each training example X^k , calculate $P(Z^k | X^k, \theta)$
 - M step: chose new θ to maximize

Learning Bayes Net Structure

How can we learn Bayes Net graph structure?

In general case, open problem

- can require lots of data (else high risk of overfitting)
- can use Bayesian methods to constrain search

One key result:

- Chow-Liu algorithm: finds “best” tree-structured network
- What’s best?
 - suppose $P(\mathbf{X})$ is true distribution, $T(\mathbf{X})$ is our tree-structured network, where $\mathbf{X} = \langle X_1, \dots, X_n \rangle$
 - Chow-Liu minimizes Kullback-Leibler divergence:

$$KL(P(\mathbf{X}) \parallel T(\mathbf{X})) \equiv \sum_k P(\mathbf{X} = k) \log \frac{P(\mathbf{X} = k)}{T(\mathbf{X} = k)}$$

Chow-Liu Algorithm

Key result: To minimize $KL(P \parallel T)$, it suffices to find the tree network T that maximizes the sum of mutual informations over its edges

Mutual information for an edge between variable A and B :

$$I(A, B) = \sum_a \sum_b P(a, b) \log \frac{P(a, b)}{P(a)P(b)}$$

This works because for tree networks with nodes $\mathbf{X} \equiv \langle X_1 \dots X_n \rangle$

$$\begin{aligned} KL(P(\mathbf{X}) \parallel T(\mathbf{X})) &\equiv \sum_k P(\mathbf{X} = k) \log \frac{P(\mathbf{X} = k)}{T(\mathbf{X} = k)} \\ &= - \sum_i I(X_i, Pa(X_i)) + \sum_i H(X_i) - H(X_1 \dots X_n) \end{aligned}$$

Chow-Liu Algorithm

1. for each pair of vars A,B, use data to estimate $P(A,B)$, $P(A)$, $P(B)$

2. for each pair of vars A,B calculate mutual information

$$I(A, B) = \sum_a \sum_b P(a, b) \log \frac{P(a, b)}{P(a)P(b)}$$

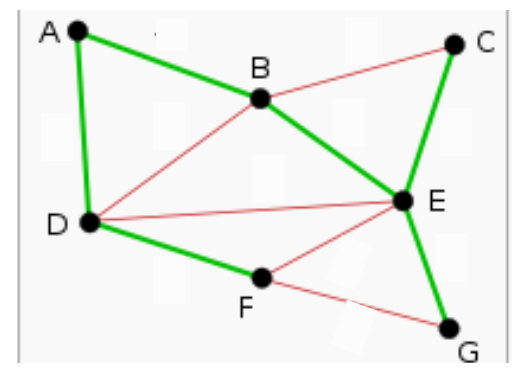
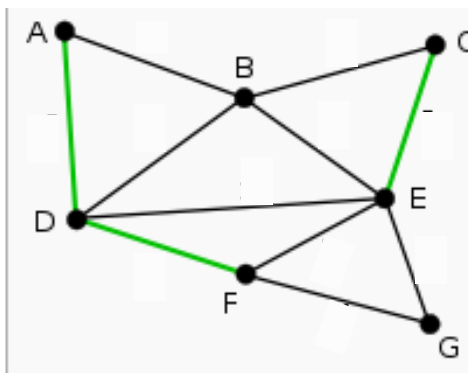
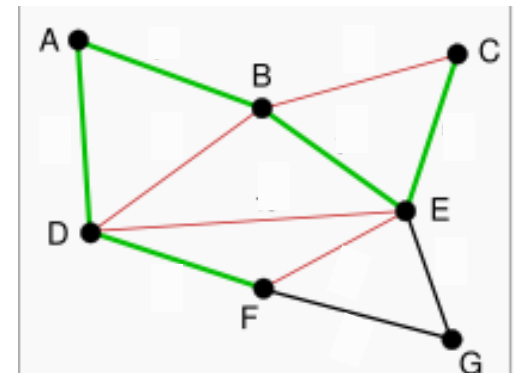
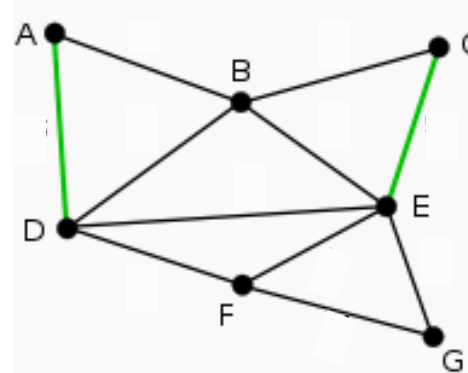
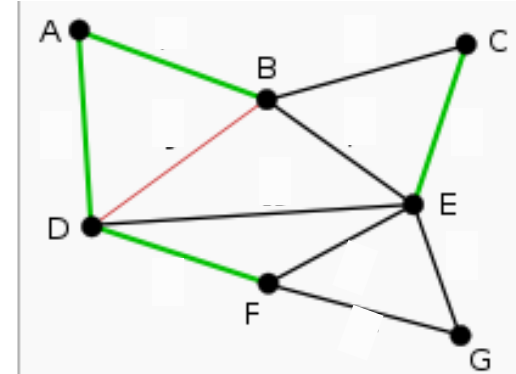
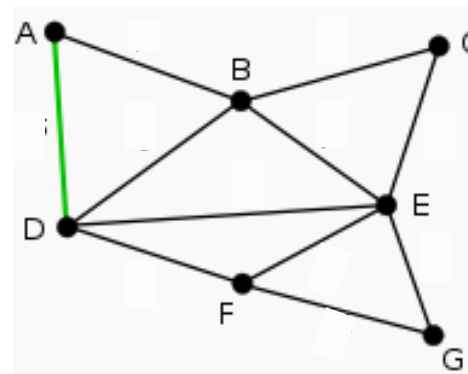
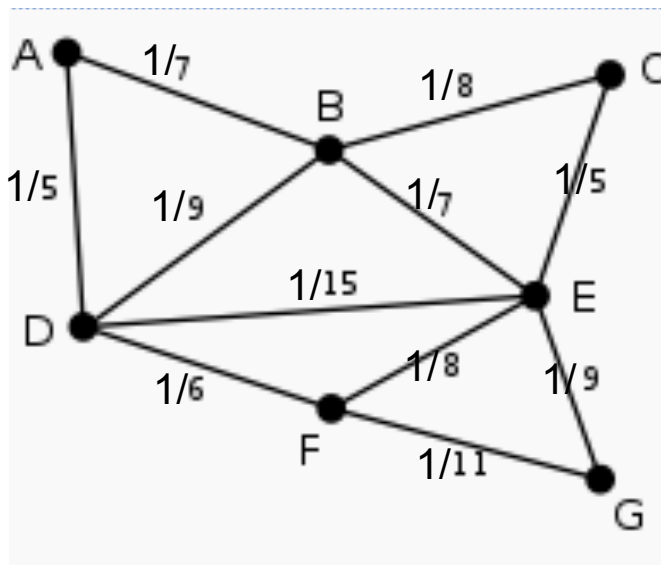
3. calculate the maximum spanning tree over the set of variables, using edge weights $I(A,B)$

(given N vars, this costs only $O(N^2)$ time)

4. add arrows to edges to form a directed-acyclic graph
5. learn the CPD's for this graph

Chow-Liu algorithm example

Greedy Algorithm to find Max-Spanning Tree



[courtesy A. Singh, C. Guestrin]

Bayes Nets – What You Should Know

- Representation
 - Bayes nets represent joint distribution as a DAG + Conditional Distributions
 - D-separation lets us decode conditional independence assumptions
- Inference
 - NP-hard in general
 - For some graphs, closed form inference is feasible
 - Approximate methods too, e.g., Monte Carlo methods, ...
- Learning
 - Easy for known graph, fully observed data (MLE's, MAP est.)
 - EM for partly observed data, known graph
 - Learning graph structure: Chow-Liu for tree-structured networks
 - Hardest when graph unknown, data incompletely observed