Bayesian Networks IV
EM, Clustering, Mixture of Gaussians
Learning network structure

Machine Learning 10-601
Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University
February 25, 2009
Bayesian Networks Definition

A Bayes network represents the joint probability distribution over a collection of random variables.

A Bayes network is a directed acyclic graph and a set of CPD’s:

- Each node denotes a random variable.
- Edges denote dependencies.
- CPD for each node $X_i$ defines $P(X_i \mid Pa(X_i))$.
- The joint distribution over all variables is defined as

$$P(X_1 \ldots X_n) = \prod_{i} P(X_i \mid Pa(X_i))$$

$Pa(X) =$ immediate parents of $X$ in the graph.
EM Algorithm

EM is a general procedure for learning from partly observed data

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

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

Iterate until convergence:

- E Step: For each training example $k$, use observed $X_k$ and current $\theta$ to calculate $P(Z_k|X_k, \theta)$

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

Guaranteed to find local maximum in $E_{P(Z|X,\theta)}[\log P(X, Z|\theta')]$
EM and estimating $\theta$

More generally, 
Given observed set $X$, unobserved set $Z$ of \textbf{boolean} random vars

Iterate E,M steps to convergence:

\begin{align*}
\text{E step: } & \text{ Calculate for each training example, } k \\
& \text{ the expected value of each unobserved variable} \\
& \text{(i.e., the probability that its value is 1)} \\
\text{M step: } & \text{ Calculate estimates similar to MLE, but} \\
& \text{ replacing each count by its expected count} \\
& \delta(Y = 1) \rightarrow E_{Z|X, \theta}[Y] \\
& \delta(Y = 0) \rightarrow (1 - E_{Z|X, \theta}[Y])
\end{align*}
Using Unlabeled Data to Help Train Naïve Bayes Classifier

Learn $P(Y|X)$

<table>
<thead>
<tr>
<th>Y</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>?</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>?</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
E step: Calculate for each training example, k
the expected value of each unobserved variable

\[ \text{Exp val for } Y_k \text{ given } X_{1k}, X_{2k}, \ldots, X_{nk} \]

\[ P(Y(\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_n|x_\theta)) = \frac{P(Y) \prod_i P(x_i|Y)}{P(x)} \]
EM and estimating $\theta$

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 $Y$

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

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

*let's use $y(k)$ to indicate value of $Y$ on kth example*
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 $Y$

$$E_{P(Y|X_1...X_N)}[y(k)] = P(y(k) = 1|X_1(k),...X_N(k); \theta) = \frac{P(y(k) = 1) \prod_{i} P(x_i(k)|y(k) = 1)}{\sum_{j=0}^{1} P(y(k) = j) \prod_{i} P(x_i(k)|y(k) = j)}$$

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

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

MLE would be:

$$\hat{P}(X_i = j|Y = m) = \frac{\sum_{k} \delta((y(k) = m) \land (x_i(k) = j))}{\sum_{k} \delta(y(k) = m)}$$
- **Inputs:** Collections $\mathcal{D}^l$ of labeled documents and $\mathcal{D}^u$ of unlabeled documents.
- Build an initial naive Bayes classifier, $\hat{\theta}$, from the labeled documents, $\mathcal{D}^l$, only. Use maximum a posteriori parameter estimation to find $\hat{\theta} = \arg \max_\theta P(\mathcal{D}|\theta)P(\theta)$ (see Equations 5 and 6).
- Loop while classifier parameters improve, as measured by the change in $\ell_c(\theta|\mathcal{D}; z)$ (the complete log probability of the labeled and unlabeled data
  - **(E-step)** Use the current classifier, $\hat{\theta}$, to estimate component membership of each unlabeled document, *i.e.*, the probability that each mixture component (and class) generated each document, $P(c_j|d_i; \hat{\theta})$ (see Equation 7).
  - **(M-step)** Re-estimate the classifier, $\hat{\theta}$, given the estimated component membership of each document. Use maximum a posteriori parameter estimation to find $\hat{\theta} = \arg \max_\theta P(\mathcal{D}|\theta)P(\theta)$ (see Equations 5 and 6).
- **Output:** A classifier, $\hat{\theta}$, that takes an unlabeled document and predicts a class label.
Experimental Evaluation

• Newsgroup postings
  – 20 newsgroups, 1000/group

• Web page classification
  – student, faculty, course, project
  – 4199 web pages

• Reuters newswire articles
  – 12,902 articles
  – 90 topics categories
Table 3. Lists of the words most predictive of the course class in the WebKB data set, as they change over iterations of EM for a specific trial. By the second iteration of EM, many common course-related words appear. The symbol $D$ indicates an arbitrary digit.

<table>
<thead>
<tr>
<th>Iteration 0</th>
<th>Iteration 1</th>
<th>Iteration 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>intelligence</td>
<td>$DD$</td>
<td>$D$</td>
</tr>
<tr>
<td>$DD$</td>
<td>$D$</td>
<td>$DD$</td>
</tr>
<tr>
<td>artificial</td>
<td>lecture</td>
<td>cc</td>
</tr>
<tr>
<td>understanding</td>
<td>$D^*$</td>
<td>$DD:DD$</td>
</tr>
<tr>
<td>$DDw$</td>
<td>due</td>
<td>due</td>
</tr>
<tr>
<td>dist</td>
<td>problem</td>
<td>set</td>
</tr>
<tr>
<td>identical</td>
<td>set</td>
<td>tay</td>
</tr>
<tr>
<td>rus</td>
<td>problem</td>
<td>$DDam$</td>
</tr>
<tr>
<td>arrange</td>
<td>due</td>
<td>homework</td>
</tr>
<tr>
<td>games</td>
<td>yurttas</td>
<td>homework</td>
</tr>
<tr>
<td>dartmouth</td>
<td>$DDam$</td>
<td>postscript</td>
</tr>
<tr>
<td>natural</td>
<td>yurttas</td>
<td>solution</td>
</tr>
<tr>
<td>cognitive</td>
<td>homework</td>
<td>quiz</td>
</tr>
<tr>
<td>logic</td>
<td>sec</td>
<td>chapter</td>
</tr>
<tr>
<td>proving</td>
<td>postscript</td>
<td>ascii</td>
</tr>
<tr>
<td>prolog</td>
<td>exam</td>
<td></td>
</tr>
<tr>
<td>knowledge</td>
<td>set</td>
<td></td>
</tr>
<tr>
<td>human</td>
<td>hw</td>
<td></td>
</tr>
<tr>
<td>representation</td>
<td>exam</td>
<td></td>
</tr>
<tr>
<td>field</td>
<td>problem</td>
<td></td>
</tr>
</tbody>
</table>

Using one labeled example per class

word $w$ ranked by $P(w|Y=$course$) / P(w|Y \neq$course$)$
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 $P(X_1, \ldots X_n)$ as mixture of multiple distributions. Use discrete-valued random var $Z$ to indicate which mixture component is being used for each random draw.

So

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

Mixture of Gaussian clustering:

- Assume each data point $X=<X_1, \ldots X_n>$ is generated by mixture of Gaussians, as follows:
  1. randomly choose a cluster $z$, according to $P(Z=z)$
  2. randomly generate a data point $<x_1,x_2 \ldots x_n>$ according to $N(\mu_z, \Sigma_z)$
EM for Mixture of Gaussian Clustering

Let’s simplify to make this easier:

1. Assume $X = <X_1 \ldots X_n>$, 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 $\sigma$ known, $\pi_1 \ldots \pi_K, \mu_{1i} \ldots \mu_{Ki}$ unknown

Observed: $X = <X_1 \ldots X_n>$

Unobserved: $Z$

Diagram:

- $Z$ is the root node.
- $X_1$, $X_2$, $X_3$, $X_4$ are the leaf nodes.
- $Z$ is connected to $X_1$, $X_2$, $X_3$, $X_4$.
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 $\theta$ by

$$\theta \leftarrow \arg \max_{\theta'} Q(\theta'|\theta)$$
EM – E Step

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

$X(n)=<x_1(n), x_2(n), \ldots x_T(n)>.$

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

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

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

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

$\pi'$ has no influence

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

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

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

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

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

$$\pi \leftarrow \frac{\sum_{n=1}^{N} E[z(n)]}{\left( \sum_{n=1}^{N} E[z(n)] \right) + \left( \sum_{n=1}^{N} (1 - E[z(n)]) \right)} = \frac{1}{N} \sum_{n=1}^{N} E[z(n)]$$
Now consider update for $\mu_{ji}$

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

$\mu_{ji}'$ has no influence

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

\[\cdots\]

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

Compare above to MLE if $Z$ were observable:

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

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

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

$$\pi \leftarrow \frac{1}{N} \sum_{n=1}^{N} E[z(n)] \quad \mu_{ji} \leftarrow \frac{\sum_{n=1}^{N} P(z(n) = j|x(n), \theta) \ x_i(n)}{\sum_{n=1}^{N} P(z(n) = j|x(n), \theta)}$$
Mixture of Gaussians applet

• Run applet
  http://www.neurosci.aist.go.jp/%7Eakaho/MixtureEM.html
What you should know about EM

• For learning from partly unobserved data
• MLEst of $\theta = \arg \max_{\theta} \log P(data|\theta)$
• 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

• 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: calculate $E_{Z|X,\theta}[\log P(X, Z|\theta)]$
  – M step: find its derivative wrt $\theta$, and set it to zero
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:
- Chou Liu algorithm: finds “best” tree-structured network
- What’s best?
  - suppose \( P(X) \) is true distribution, \( T(X) \) is our tree-structured network
  - minimize Kullback-Leibler divergence:

\[
KL(P(X), T(X)) = \sum_i P(X = x_i) \log \frac{P(X = x_i)}{T(X = x_i)}
\]
Chou-Liu Algorithm

Key result:
To minimize $KL(P,T)$, the tree network must be a tree whose edges maximize the total mutual information

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

i.e., tree such that sum of $I(X,Y)$ over all edges is maximum

$$KL(P(X), T(X)) = \sum_{i} P(X = x_i) \log \frac{P(X = x_i)}{T(X = x_i)}$$
Chou-Liu Algorithm

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

2. for each pair of vars A, B calculate

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

3. calculate the maximum spanning tree over the set of variables (given M vars, this costs only $O(M^2)$ time)

4. add arrows to the edges to form a directed-acyclic graph

5. learn the CPD’s for this graph