430.457

Introduction to Intelligent Systems

Prof. Songhwai Oh

ECE, SNU

LEARNING WITH HIDDEN VARIABLES: THE EM ALGORITHM

Latent (or Hidden) Variables

- Latent variables can dramatically reduce the number of parameters required to specify a Bayesian network.

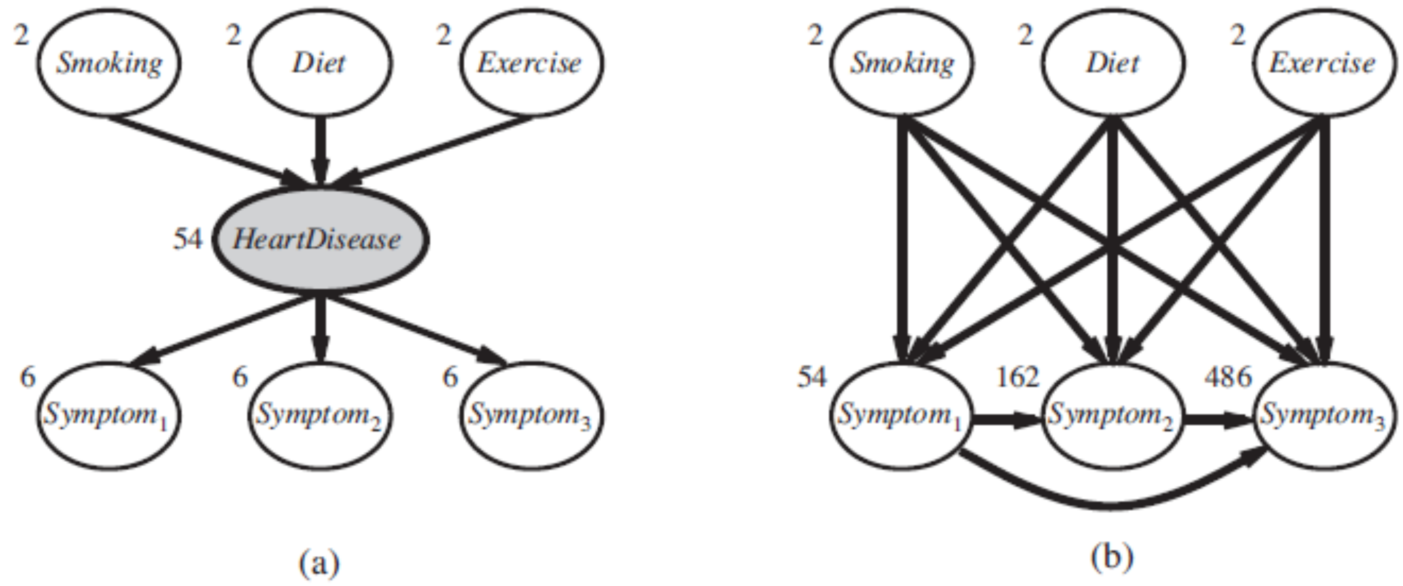
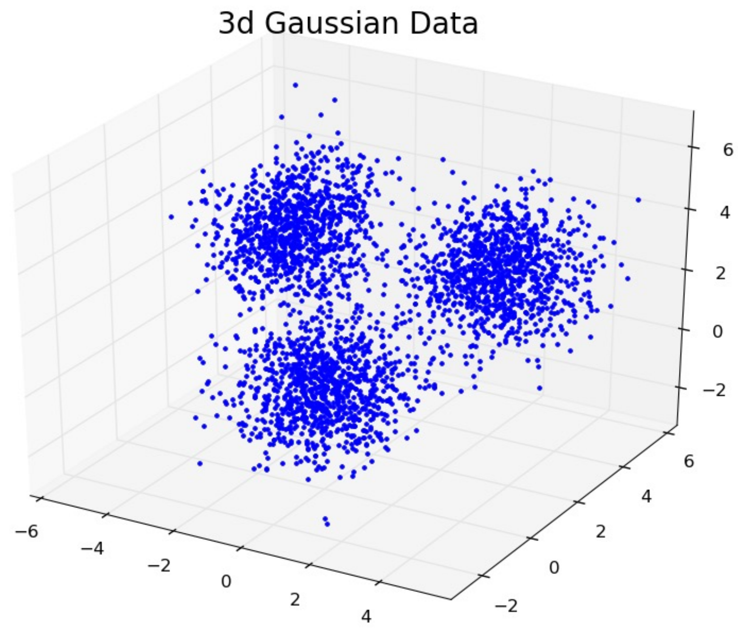


Figure 20.10 (a) A simple diagnostic network for heart disease, which is assumed to be a hidden variable. Each variable has three possible values and is labeled with the number of independent parameters in its conditional distribution; the total number is 78. (b) The equivalent network with *HeartDisease* removed. Note that the symptom variables are no longer conditionally independent given their parents. This network requires 708 parameters.

Unsupervised Clustering

- The problem of discerning multiple categories in a collection of objects.
- The problem is unsupervised because the category labels are not given.
- **Chicken-and-egg problem**: We do not know the assignments nor the parameters



k-means Algorithm

- $k = 2$ (number of clusters)
- Means: μ_1, μ_2
- Indicator variables $c_n^i \in \{0, 1\}$: $c_n^i = 1$ if x_n is assigned to the ith cluster.
- Iterate the following two steps until convergence.

  1. Find assignments

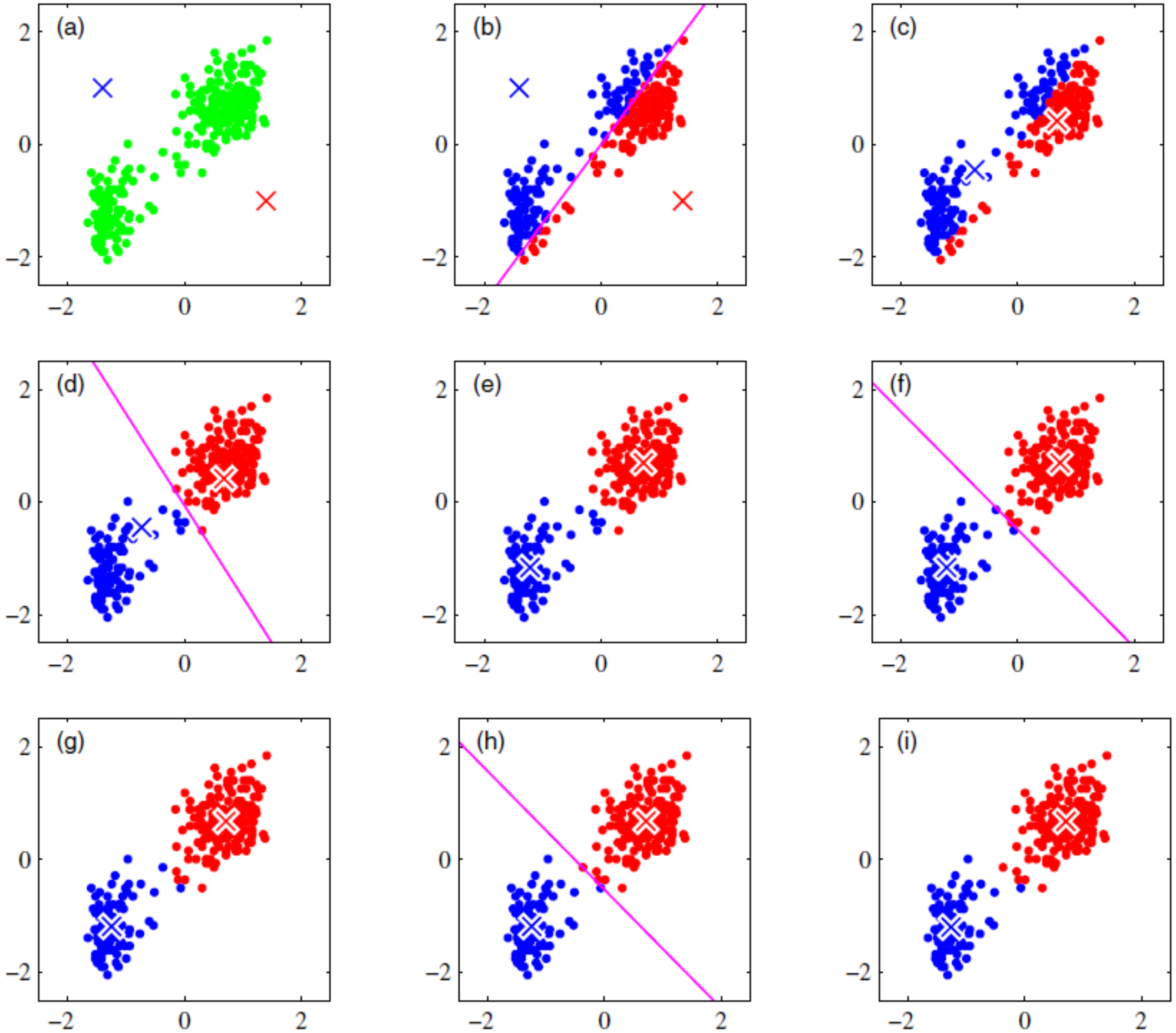
$$c_n^i = \begin{cases} 1 & \text{if } i = \arg\min_j \|x_n - \mu_j\|^2 \\ 0 & \text{otherwise.} \end{cases}$$

  2. Update means

$$\mu_i = \frac{\sum_n c_n^i x_n}{\sum_n c_n^i}$$

- k-means algorithm finds a solution which minimizes the following cost function (distortion measure).

$$J = \sum_{n=1}^{N} \sum_{i=1}^{k} c_n^i \|x_n - \mu_i\|^2.$$


(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)

Mixture Models

- Mixture distribution with k components:

mixture weight

$$P(\mathbf{x}) = \sum_{i=1}^{k} P(C=i) P(\mathbf{x}|C=i)$$

- Mixture of Gaussians (or a Gaussian Mixture Model (GMM))

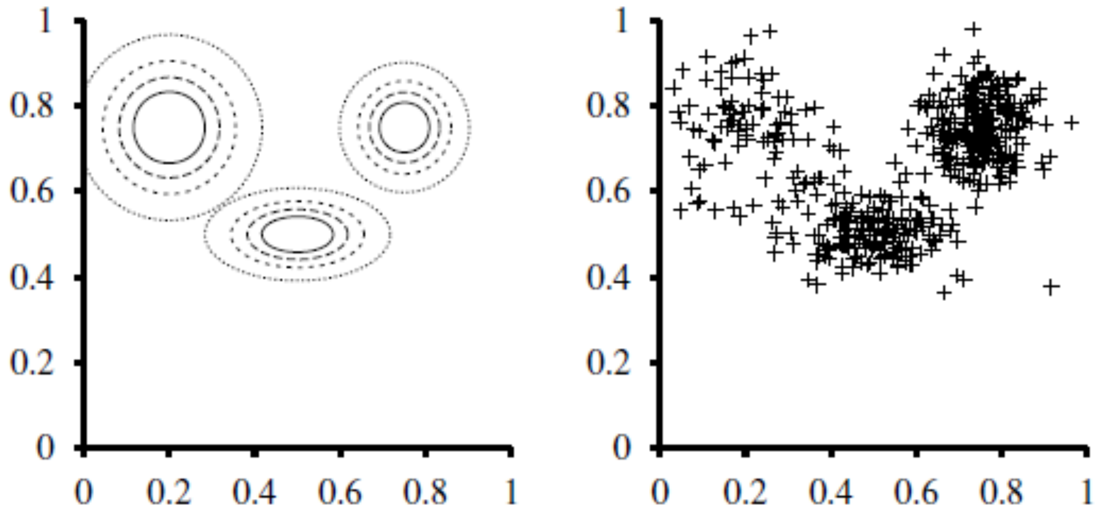
$$P(\mathbf{x}) = \sum_{i=1}^{k} P(C=i) \mathcal{N}(\mathbf{x}|\mu_i, \mathbf{\Sigma}_i)$$

- Mixture of Gaussians

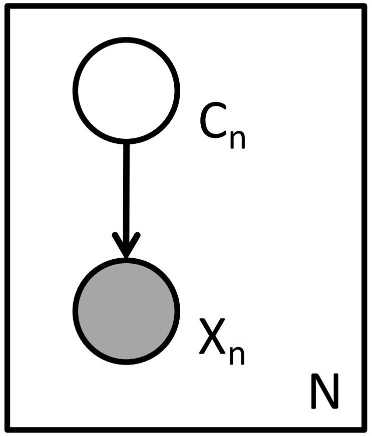
$$P(\mathbf{x}) = \sum_{i=1}^{k} P(C=i)\mathcal{N}(\mathbf{x}|\mu_i, \mathbf{\Sigma}_i)$$

Generative model

1. Choose the component with probability P(C=i)
2. Generate a sample using the distribution of the chosen component

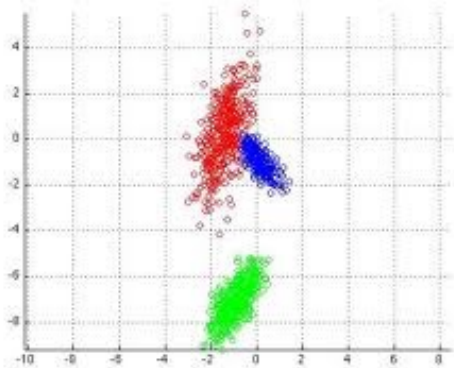


Mixture Models



$$
\begin{aligned}
C_n &= \text{multinomial}(K; \pi) \\
P(X_n|\theta) &= \sum_{i=1}^{K} P(C_n^i = 1|\pi) P(X_n|C_n^i = 1, \theta_i)
\end{aligned}
$$

Mixture of Gaussians



$$P(x_n|c_n^i = 1, \theta_i) = \mathcal{N}(x_n|\mu_i, \Sigma_i)$$

Likelihood $$P(x_1, \ldots, x_N|\theta) = \prod_{n=1}^{N} \left(\sum_{i=1}^{K} P(C_n^i = 1|\pi) P(x_n|C_n^i = 1, \theta_i) \right)$$

Log-likelihood $$\mathcal{L}(\theta|x_1, \ldots, x_N) = \sum_{n=1}^{N} \log \left\{ \sum_{i=1}^{K} P(C_n^i = 1|\pi) P(x_n|C_n^i = 1, \theta_i) \right\}$$

No closed-form ML solution

Expectation-Maximization (EM) Algorithm

For the mixture of Gaussians, we initialize the mixture-model parameters arbitrarily and then iterate the following two steps:

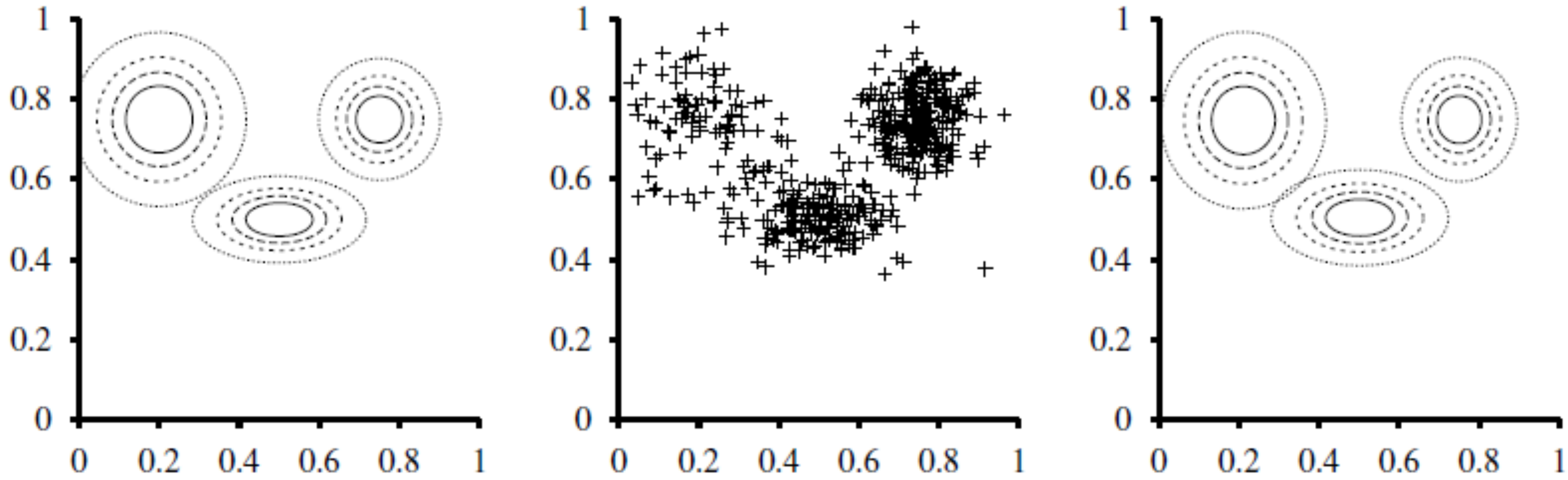
1. **E-step**: Compute the probabilities $p_{ij} = P(C = i \mid \mathbf{x}_j)$, the probability that datum $\mathbf{x}_j$ was generated by component i. By Bayes' rule, we have $p_{ij} = \alpha P(\mathbf{x}_j \mid C = i) P(C = i)$. The term $P(\mathbf{x}_j \mid C = i)$ is just the probability at $\mathbf{x}_j$ of the ith Gaussian, and the term $P(C = i)$ is just the weight parameter for the ith Gaussian. Define $n_i = \sum_j p_{ij}$, the effective number of data points currently assigned to component i.
2. **M-step**: Compute the new mean, covariance, and component weights using the following steps in sequence:

$$\boldsymbol{\mu}_i \leftarrow \sum_j p_{ij} \mathbf{x}_j / n_i$$

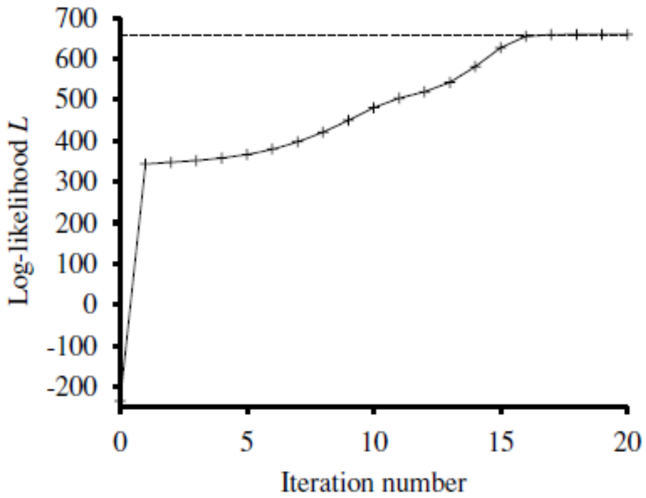
$$\boldsymbol{\Sigma}_i \leftarrow \sum_j p_{ij} (\mathbf{x}_j - \boldsymbol{\mu}_i)(\mathbf{x}_j - \boldsymbol{\mu}_i)^\top / n_i$$

$$w_i \leftarrow n_i / N$$

E-step (expectation step) computes the expected values p_{ij} of the hidden indicator variables Z_{ij}, where $Z_{ij} = 1$ if x_j was generated by the ith component and 0 otherwise. M-step (maximization step) finds the ML estimates, given the expected values of the hidden indicator variables.

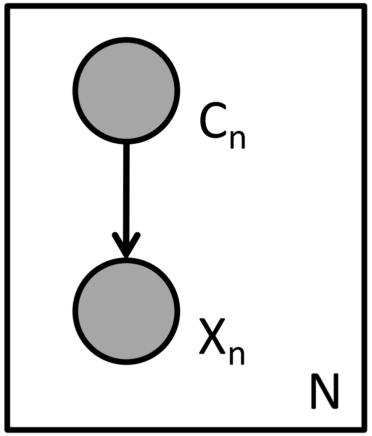


The model learned by the EM algorithm



- The EM algorithm increases the log likelihood at every iteration
- The EM algorithm converges to a local maximum in likelihood.

EM Algorithm



Pretend all nodes are observed

$$\mathcal{D}_c = \{(x_n, c_n) : n = 1, \ldots, N\}$$

Complete likelihood

$$\prod_{n=1}^{N} \prod_{i=1}^{K} \left(P(c_n^i = 1 | \pi) P(x_n | c_n^i = 1, \theta_i) \right)^{c_n^i}$$

Complete log-likelihood

$$\mathcal{L}_c(\theta | \mathcal{D}_c) = \sum_{n=1}^{N} \sum_{i=1}^{K} c_n^i \log \left(P(c_n^i = 1 | \pi) P(x_n | c_n^i = 1, \theta_i) \right)$$

Expected complete log-likelihood

$$\begin{aligned} \mathbb{E}_{\tilde{\theta}} \left(\mathcal{L}_c(\theta | \mathcal{D}_c) \right) &= \mathbb{E}_{\tilde{\theta}} \left(\sum_{n=1}^{N} \sum_{i=1}^{K} c_n^i \log \left(P(c_n^i = 1 | \pi) P(x_n | c_n^i = 1, \theta_i) \right) \right) \\ &= \sum_{n=1}^{N} \sum_{i=1}^{K} \mathbb{E}_{\tilde{\theta}} \left(c_n^i \right) \log \left(P(c_n^i = 1 | \pi) P(x_n | c_n^i = 1, \theta_i) \right) \end{aligned}$$

EM Algorithm

1. E-step: Compute $\mathbb{E}_{\theta^{(t)}}\left(c_n^i\right) = P(c_n^i = 1 | x_1, \ldots, x_N, \theta^{(t)})$
2. M-step: Maximize $\mathbb{E}_{\theta^{(t)}}\left(\mathcal{L}_c(\theta|\mathcal{D}_c)\right)$ with respect to θ; the solution becomes $\theta^{(t+1)}$.
3. Iterate until it converges