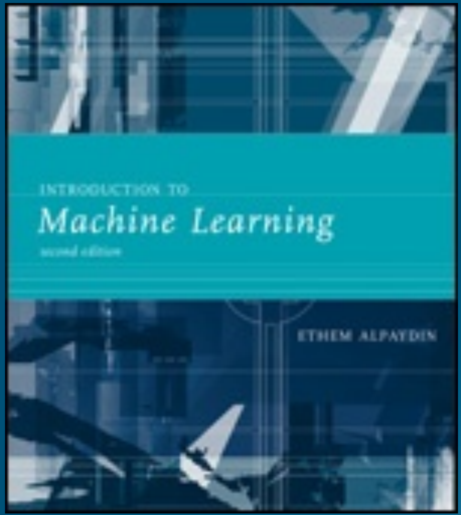


Lecture Slides for

INTRODUCTION TO

Machine Learning

2nd Edition

ETHEM ALPAYDIN

alpaydin@boun.edu.tr
http://www.cmpe.boun.edu.tr/~ethem/i2ml2e

CHAPTER 7:

Clustering

Semiparametric Density Estimation

- Parametric: Assume a single model for $p(\boldsymbol{x} \mid C_i)$ (Chapters 4 and 5)
- Semiparametric: $p(\boldsymbol{x} \mid C_i)$ is a mixture of densities

  Multiple possible explanations/prototypes:

  Different handwriting styles, accents in speech
- Nonparametric: No model; data speaks for itself (Chapter 8)

Mixture Densities

$$p(\mathbf{x}) = \sum_{i=1}^{k} p(\mathbf{x} \mid G_i) P(G_i)$$

where G_i the components/groups/clusters,

$P(G_i)$ mixture proportions (priors),

$p(\boldsymbol{x} \mid G_i)$ component densities

Gaussian mixture where $p(\boldsymbol{x}|G_i) \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ parameters $\Phi = \{P(G_i), \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^{k}$

unlabeled sample $X=\{\boldsymbol{x}^t\}_t$ (unsupervised learning)

Classes vs. Clusters

- Supervised: X = { $\boldsymbol{x}^t, \boldsymbol{r}^t$ }$_t$
- Classes C_i $i=1,\ldots,K$

$$p(\mathbf{x}) = \sum_{i=1}^{K} p(\mathbf{x} \mid C_i) P(C_i)$$

where $p(\boldsymbol{x} \mid C_i) \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$

- $\Phi = \{P(C_i), \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^{K}$

$$\hat{P}(C_i) = \frac{\sum_t r_i^t}{N} \quad \mathbf{m}_i = \frac{\sum_t r_i^t \mathbf{x}^t}{\sum_t r_i^t}$$

$$\mathbf{S}_i = \frac{\sum_t r_i^t (\mathbf{x}^t - \mathbf{m}_i)(\mathbf{x}^t - \mathbf{m}_i)^T}{\sum_t r_i^t}$$

- Unsupervised : X = { $\boldsymbol{x}^t$ }$_t$
- Clusters G_i $i=1,\ldots,k$

$$p(\mathbf{x}) = \sum_{i=1}^{k} p(\mathbf{x} \mid G_i) P(G_i)$$

where $p(\boldsymbol{x} \mid G_i) \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$

- $\Phi = \{P(G_i), \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i\}_{i=1}^{k}$

Labels, $\boldsymbol{r}_i^t$?

k-Means Clustering

- Find k reference vectors (prototypes/codebook vectors/codewords) which best represent data
- Reference vectors, $\boldsymbol{m}_j$, j =1,...,k
- Use nearest (most similar) reference:

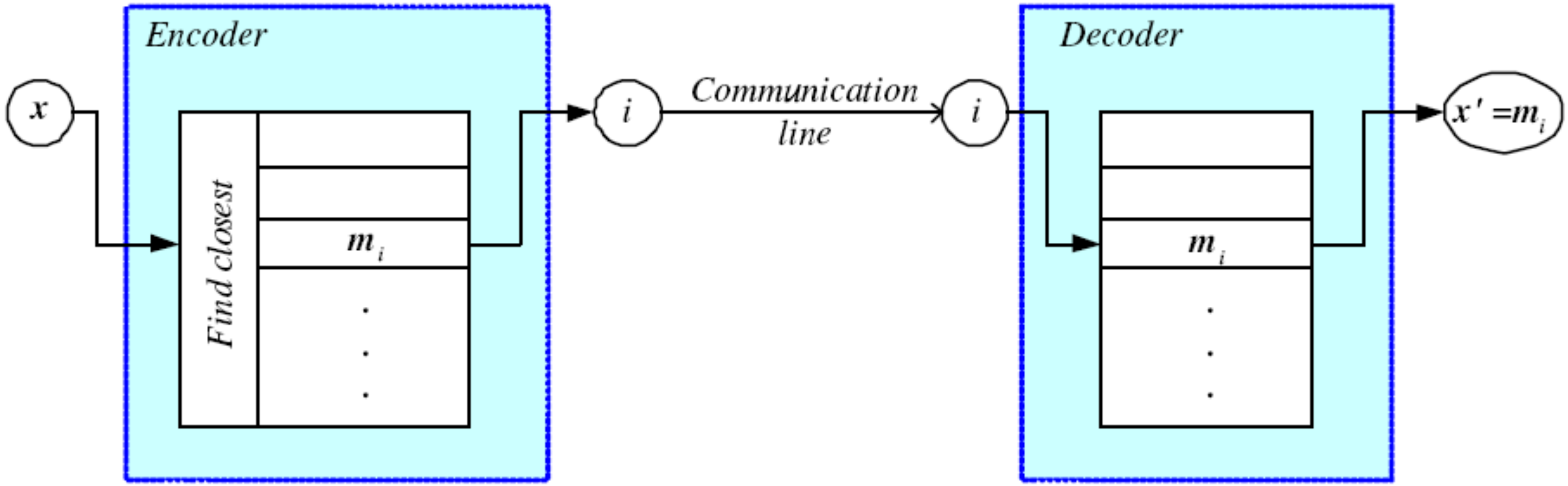
$$\left\|\mathbf{x}^t - \mathbf{m}_i\right\| = \min_j \left\|\mathbf{x}^t - \mathbf{m}_j\right\|$$

- Reconstruction error

$$E\left(\{\mathbf{m}_i\}_{i=1}^k \middle| \mathcal{X}\right) = \sum_t \sum_i b_i^t \left\|\mathbf{x}^t - \mathbf{m}_i\right\|$$

$$b_i^t = \begin{cases} 1 & \text{if } \left\|\mathbf{x}^t - \mathbf{m}_i\right\| = \min_j \left\|\mathbf{x}^t - \mathbf{m}_j\right\| \\ 0 & \text{otherwise} \end{cases}$$

Encoding/Decoding



k-means Clustering

Initialize $\boldsymbol{m}_i, i = 1, \ldots, k$, for example, to k random $\boldsymbol{x}^t$

Repeat

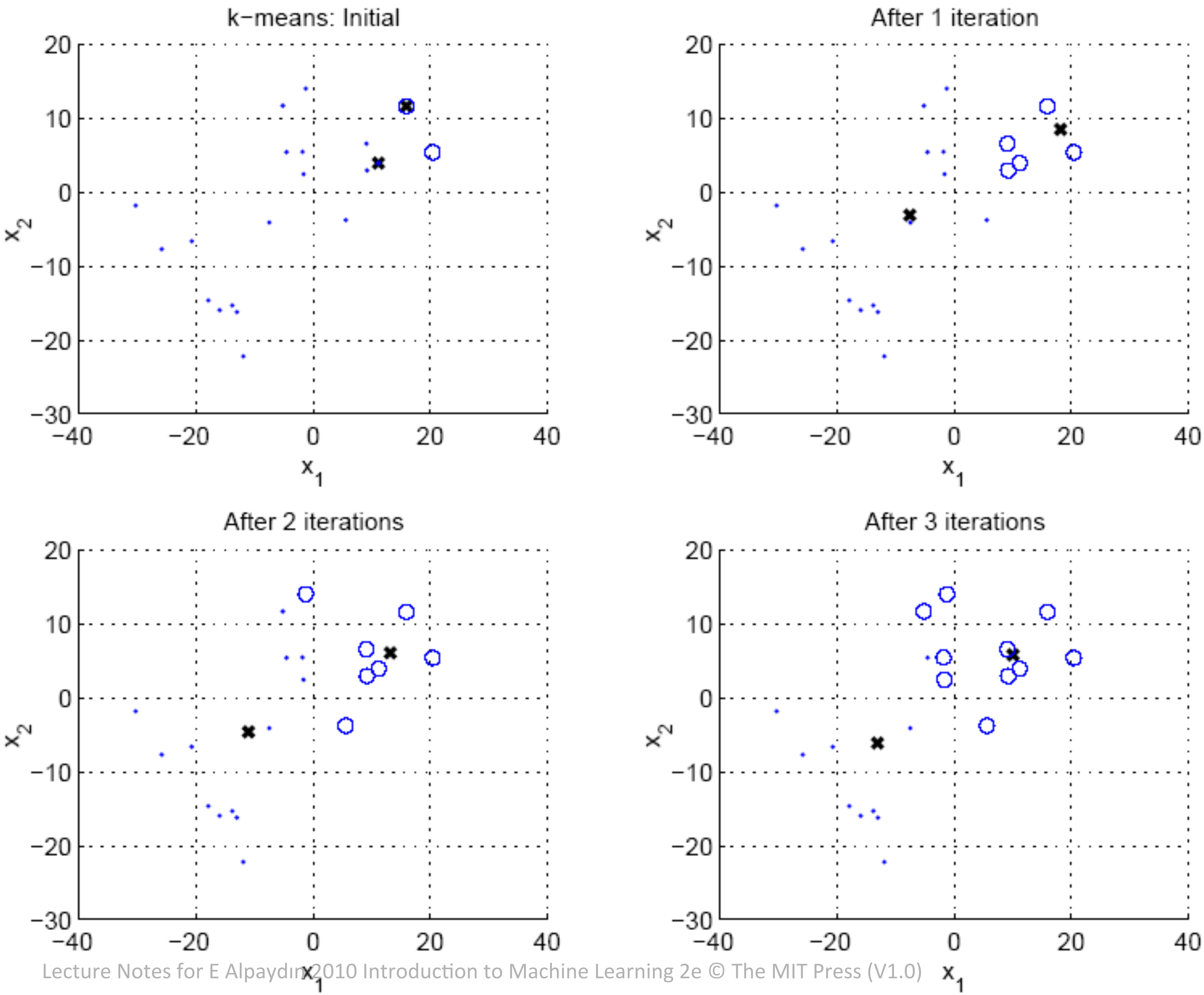
For all $\boldsymbol{x}^t \in \mathcal{X}$

$$b_i^t \leftarrow \begin{cases} 1 & \text{if } \|\boldsymbol{x}^t - \boldsymbol{m}_i\| = \min_j \|\boldsymbol{x}^t - \boldsymbol{m}_j\| \\ 0 & \text{otherwise} \end{cases}$$

For all $\boldsymbol{m}_i, i = 1, \ldots, k$

$$\boldsymbol{m}_i \leftarrow \sum_t b_i^t \boldsymbol{x}^t / \sum_t b_i^t$$

Until $\boldsymbol{m}_i$ converge


k-means: Initial
After 1 iteration
After 2 iterations
After 3 iterations
x_1
x_2

Expectation-Maximization (EM)

- Log likelihood with a mixture model

$$\mathcal{L}(\Phi \mid \mathcal{X}) = \log \prod_t p(\mathbf{x}^t \mid \Phi)$$

$$= \sum_t \log \sum_{i=1}^{k} p(\mathbf{x}^t \mid G_i) P(G_i)$$

- Assume hidden variables *z*, which when known, make optimization much simpler
- Complete likelihood, $L_c(\Phi \mid X,Z)$, in terms of ***x*** and ***z***
- Incomplete likelihood, $L(\Phi \mid X)$, in terms of ***x***

E- and M-steps

- Iterate the two steps

1. E-step: Estimate *z* given X and current Φ
2. M-step: Find new Φ' given *z*, X, and old Φ.

$$\text{E-step}: \mathcal{Q}\left(\Phi \mid \Phi^l\right) = E\left[\mathcal{L}_C\left(\Phi \mid \mathcal{X}, \mathcal{Z}\right) \mid \mathcal{X}, \Phi^l\right]$$

$$\text{M-step}: \Phi^{l+1} = \arg\max_{\Phi} \mathcal{Q}\left(\Phi \mid \Phi^l\right)$$

An increase in Q increases incomplete likelihood

$$\mathcal{L}\left(\Phi^{l+1} \mid \mathcal{X}\right) \geq \mathcal{L}\left(\Phi^l \mid \mathcal{X}\right)$$

EM in Gaussian Mixtures

- $z^t_i = 1$ if $\boldsymbol{x}^t$ belongs to G_i, 0 otherwise (labels $\boldsymbol{r}^t_i$ of supervised learning); assume $p(\boldsymbol{x}|G_i) \sim N(\boldsymbol{\mu}_i, \Sigma_i)$
- E-step:

$$E\left[z_i^t \middle| \mathcal{X}, \Phi^l\right] = \frac{p(\mathbf{x}^t \mid G_i, \Phi^l) P(G_i)}{\sum_j p(\mathbf{x}^t \mid G_j, \Phi^l) P(G_j)}$$

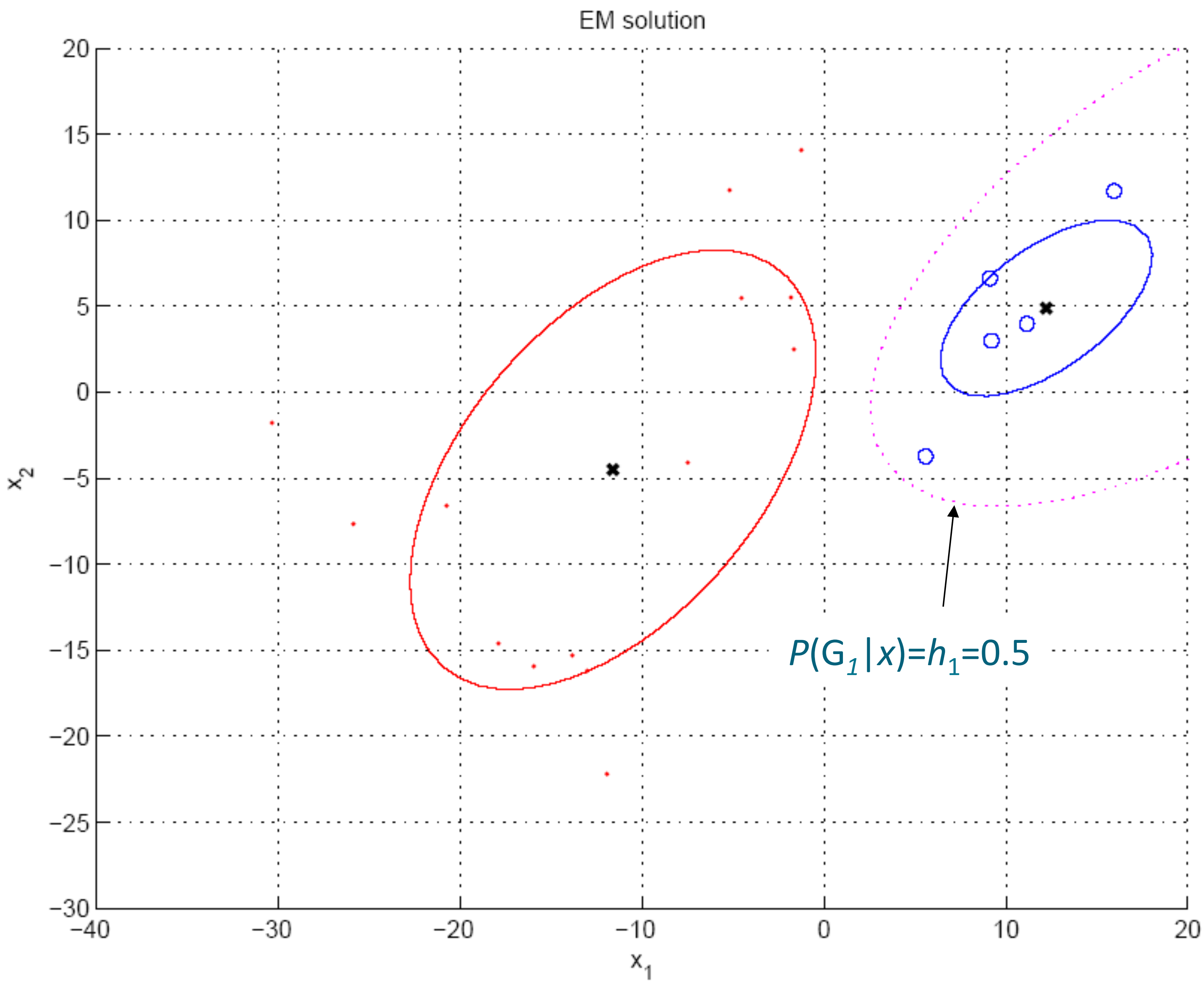
$$= P(G_i \mid \mathbf{x}^t, \Phi^l) \equiv h_i^t$$

- M-step:

$$P(G_i) = \frac{\sum_t h_i^t}{N} \qquad \mathbf{m}_i^{l+1} = \frac{\sum_t h_i^t \mathbf{x}^t}{\sum_t h_i^t}$$

Use estimated soft labels h_i^t in place of unknown labels r_i^t

$$\mathbf{S}_i^{l+1} = \frac{\sum_t h_i^t (\mathbf{x}^t - \mathbf{m}_i^{l+1})(\mathbf{x}^t - \mathbf{m}_i^{l+1})^T}{\sum_t h_i^t}$$



EM solution
20
15
10
5
0
−5
−10
−15
−20
−25
−30
−40
−30
−20
−10
0
10
20
x_1
x_2
$P(G_1|x)=h_1=0.5$

Mixtures of Latent Variable Models

- Regularize clusters

1. Assume shared/diagonal covariance matrices
2. Use PCA/FA to decrease dimensionality: Mixtures of PCA/FA

$$p(\mathbf{x}_t \mid G_i) = \mathcal{N}\left(\mathbf{m}_i, \mathbf{V}_i \mathbf{V}_i^T + \boldsymbol{\psi}_i\right)$$

Can use EM to learn $\mathbf{V}_i$ (Ghahramani and Hinton, 1997; Tipping and Bishop, 1999)

After Clustering

- Dimensionality reduction methods find correlations between features and group features
- Clustering methods find similarities between instances and group instances
- Allows knowledge extraction through

  number of clusters,

  prior probabilities,

  cluster parameters, i.e., center, range of features.

  Example: CRM, customer segmentation

Clustering as Preprocessing

- Estimated group labels h_j (soft) or b_j (hard) may be seen as the dimensions of a new k dimensional space, where we can then learn our discriminant or regressor.
- Local representation (only one b_j is 1, all others are 0; only few h_j are nonzero) vs

  Distributed representation (After PCA; all z_j are nonzero)

Mixture of Mixtures

- In classification, the input comes from a mixture of classes (supervised).
- If each class is also a mixture, e.g., of Gaussians, (unsupervised), we have a mixture of mixtures:

$$p(\mathbf{x}|C_i) = \sum_{j=1}^{k_i} p(\mathbf{x}|G_{ij})P(G_{ij})$$

$$p(\mathbf{x}) = \sum_{i=1}^{K} p(\mathbf{x}|C_i)P(C_i)$$

Hierarchical Clustering

- Cluster based on similarities/distances
- Distance measure between instances $\boldsymbol{x}^r$ and $\boldsymbol{x}^s$

  Minkowski (L_p) (Euclidean for p = 2)

$$d_m\left(\mathbf{x}^r, \mathbf{x}^s\right) = \left[\sum_{j=1}^{d}\left(x_j^r - x_j^s\right)^p\right]^{1/p}$$

  City-block distance

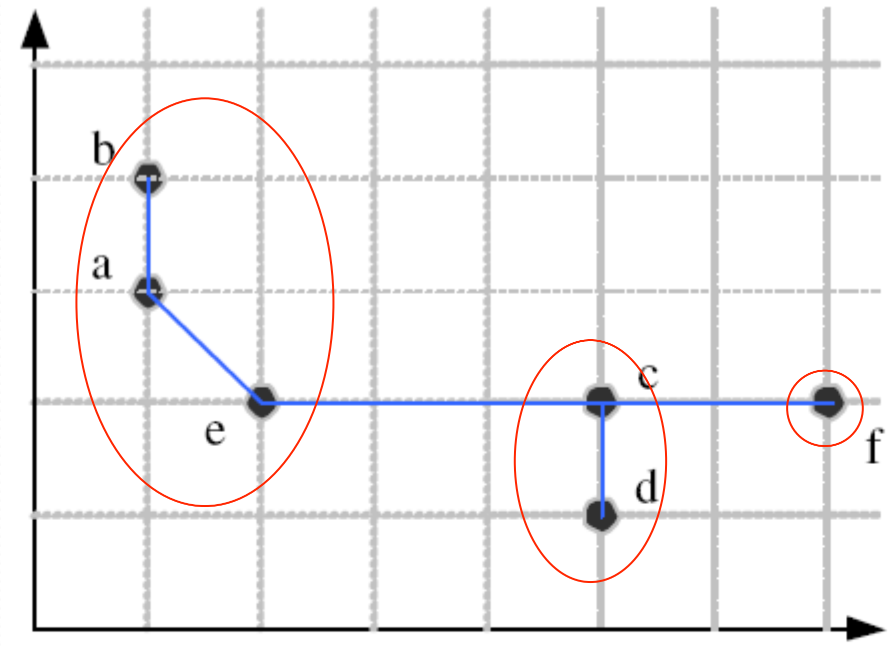
$$d_{cb}\left(\mathbf{x}^r, \mathbf{x}^s\right) = \sum_{j=1}^{d}\left|x_j^r - x_j^s\right|$$

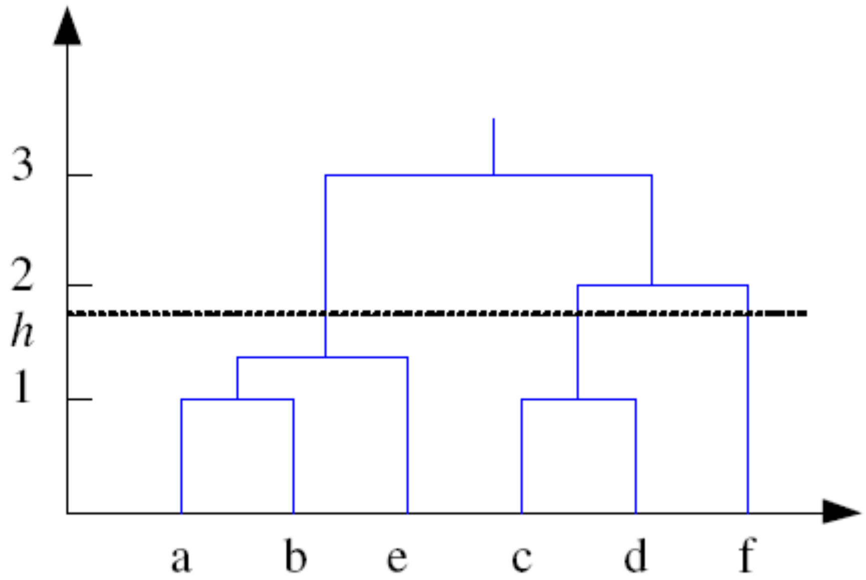
Agglomerative/Divisive Clustering

- Agglomerative: Start with N groups each with one instance and merge two closest groups at each iteration
- Divisive: Start with 1 groups containing all instances and divide large groups into smaller groups at each iteration using a flat clustering algorithm such as kmeans.

- Distance between two groups G_i and G_j:
  - Single-link: $$d(G_i, G_j) = \min_{\mathbf{x}^r \in \mathcal{G}_i, \mathbf{x}^s \in \mathcal{G}_j} d(\mathbf{x}^r, \mathbf{x}^s)$$
  - Complete-link: $$d(G_i, G_j) = \max_{\mathbf{x}^r \in \mathcal{G}_i, \mathbf{x}^s \in \mathcal{G}_j} d(\mathbf{x}^r, \mathbf{x}^s)$$
  - Average-link, distance between centroids

Example: Single-Link Clustering





Dendrogram

Choosing *k*

- Defined by the application, e.g., image quantization
- Plot data (after PCA) and check for clusters
- Incremental (leader-cluster) algorithm: Add one at a time until “elbow” (reconstruction error/log likelihood/ intergroup distances)
- Manually check for meaning