CHAPTER 7:

Clustering

Clustering

- Unsupervised classification

- Some Important Applications
  - Categorization
  - Visualization
  - Preprocessing

Clustering: Types

- Exclusive
  - One object belongs to exactly one cluster
- Overlapping
  - An object may belong to more than one cluster
- Fuzzy
  - Membership weights, between 0 and 1.

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

- $\Phi = \{P(C_i), \boldsymbol{\mu}_i, \boldsymbol{\sum}_i\}^K_{i=1}$

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

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

- **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 $p(\boldsymbol{x} \mid G_i) \sim N(\boldsymbol{\mu}_i, \boldsymbol{\sum}_i)$

- $\Phi = \{P(G_i), \boldsymbol{\mu}_i, \boldsymbol{\sum}_i\}^k_{i=1}$

Labels, $\boldsymbol{r}^t_i$?

k-Means Clustering

- Partition clustering: non-overlapping clusters
- A variation: K-medoids (use median instead of mean)
- Find *k* reference vectors (prototypes/codebook vectors/codewords) which best represent data
- Reference vectors, $\boldsymbol{m}_j$, $j = 1,\ldots,k$
- Use nearest (most similar) reference:

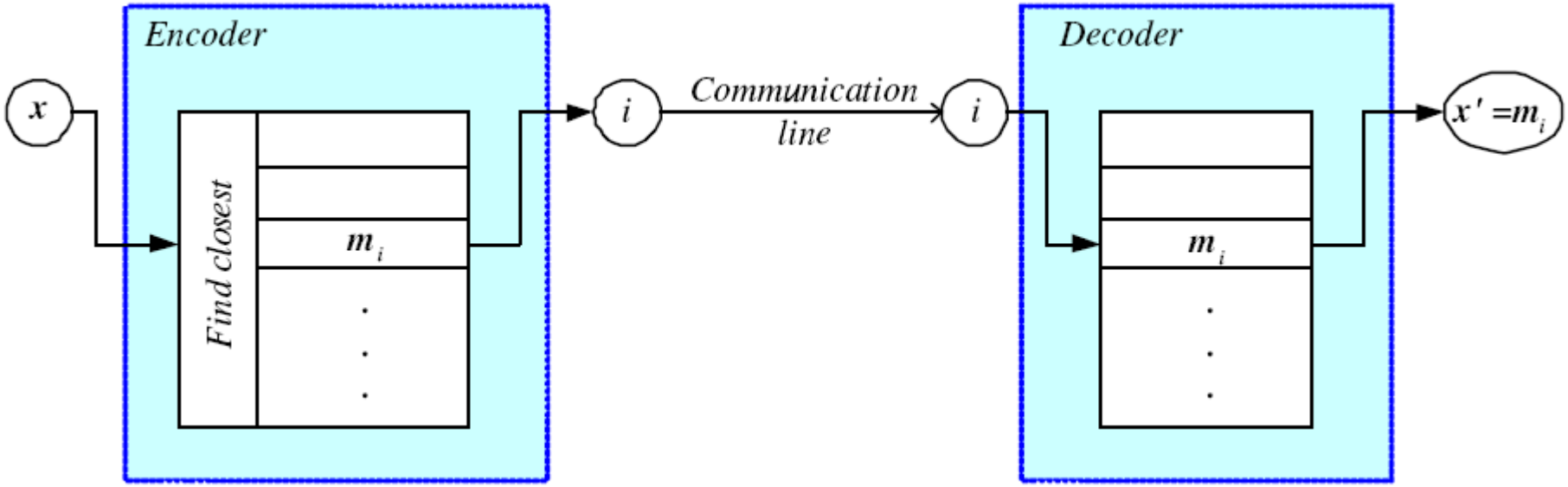
$$\left\|\mathbf{x}^t - \mathbf{m}_i\right\| = \min_j \left\|\mathbf{x}^t - \mathbf{m}_j\right\|$$

- Reconstruction error

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

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

Encoding/Decoding



k-means Clustering Algorithm

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

Repeat

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

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

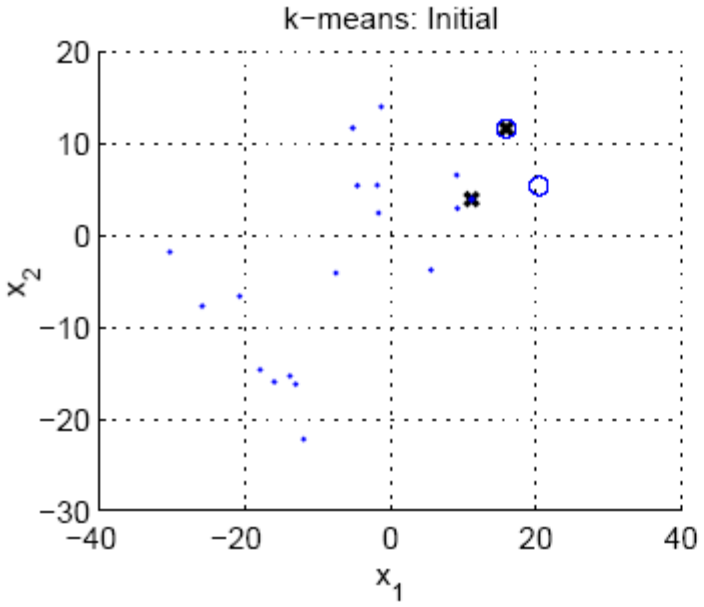
For all $\boldsymbol{m}_i, i = 1, \ldots, k$

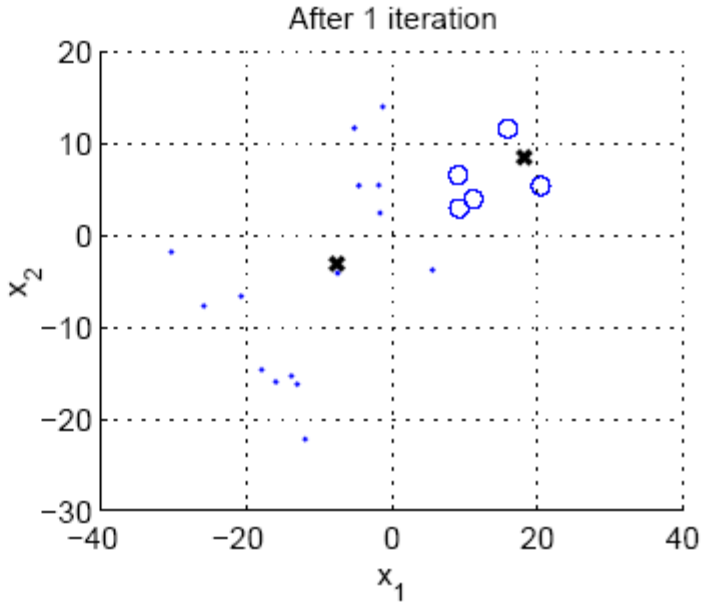
$$\boldsymbol{m}_i \leftarrow \sum_t b_i^t \boldsymbol{x}^t / \sum_t b_i^t$$

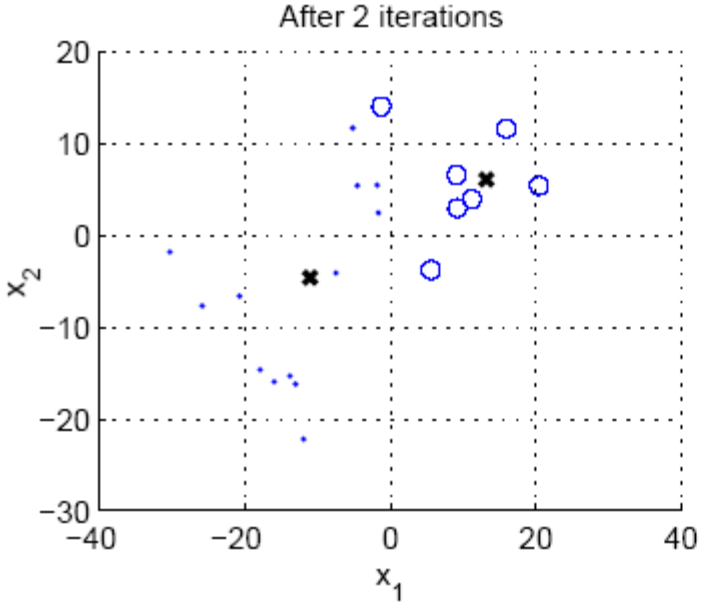
Until $\boldsymbol{m}_i$ converge

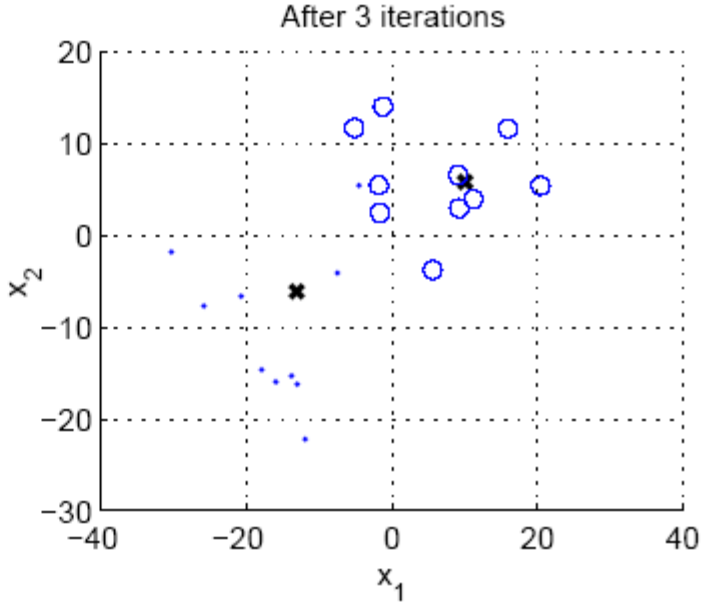
K-means clustering: other ways to initialize

- Use a fraction of data (10%)
- Find centroids
- Use those centroids for the whole data.


k−means: Initial
20
10
0
−10
−20
−30
−40
−20
0
20
40
x_2
x_1


After 1 iteration
20
10
0
−10
−20
−30
−40
−20
0
20
40
x_2
x_1


After 2 iterations
20
10
0
−10
−20
−30
−40
−20
0
20
40
x_2
x_1


After 3 iterations
20
10
0
−10
−20
−30
−40
−20
0
20
40
x_2
x_1

K-means clustering: Pros and cons

- Pros
  - Efficient algorithm
  - Widely application to various types of data

- Cons
  - Not suitable for all types of data
  - Initialization problems and outliers
  - restricted to data in which there is a notion of a center

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

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

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

Clustering using mixture models

- Each distribution corresponds to a cluster

- Find the parameters for each distribution

Expectation –Maximization (EM)

Select an initial set of model parameters.

Repeat

Expectation Step

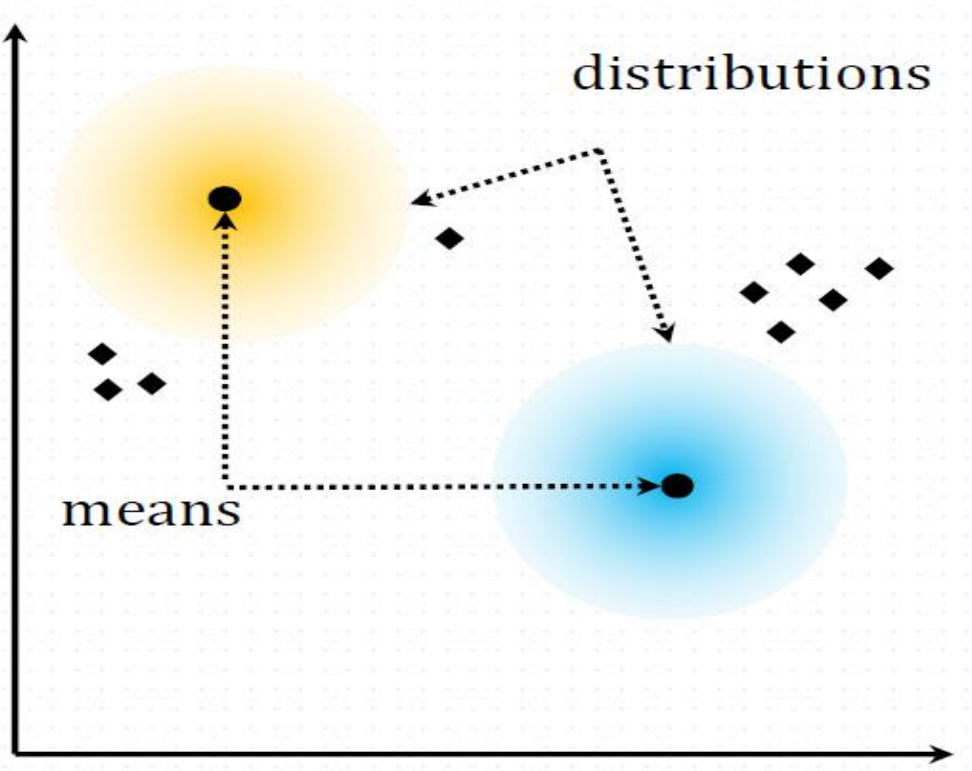
For each object, calculate probability that each object belongs to each distribution,

Maximization Step

Given the probabilities from the expectation step, find the new estimates of the parameters that maximize the expected likelihood

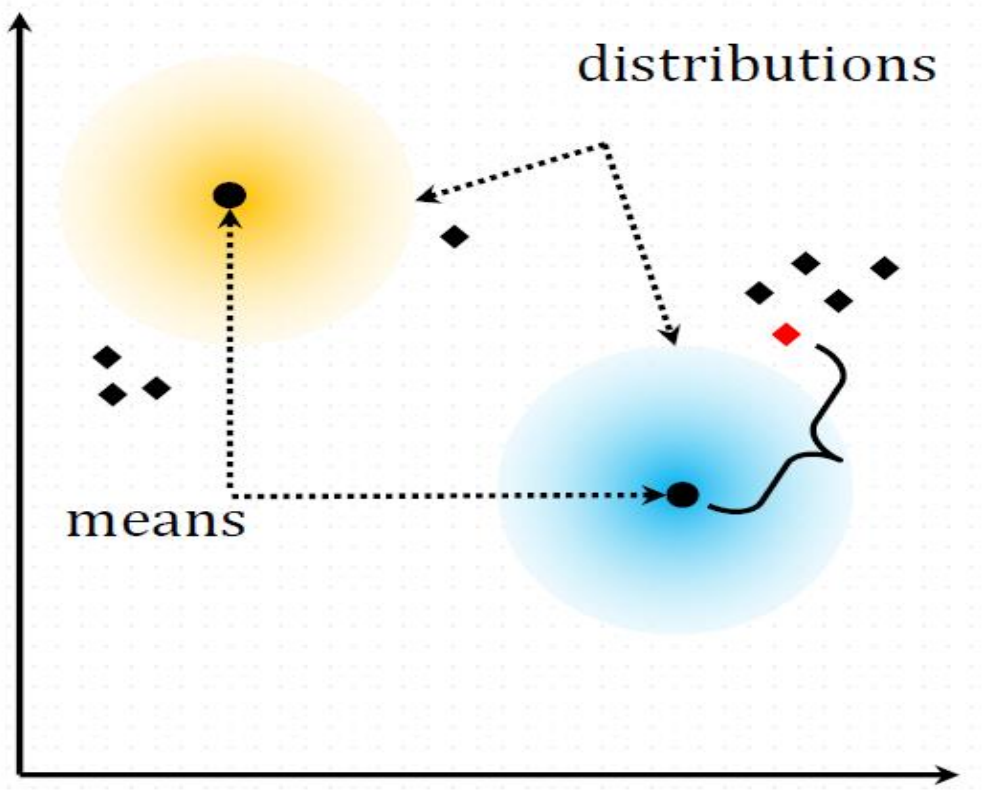
Until The parameters do not change (or are below a threshold)

EM: Visual Idea: Initialize



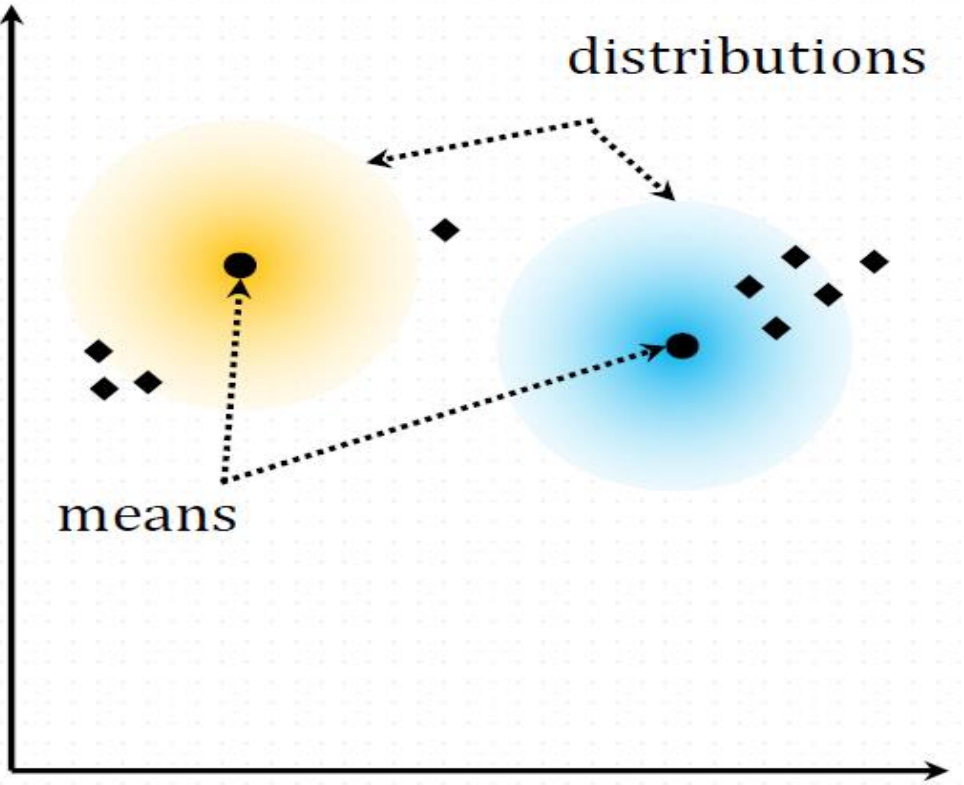
- Make an initial guess for means (initialize varances to some fixed value)

EM: Visual Idea: Expectation step



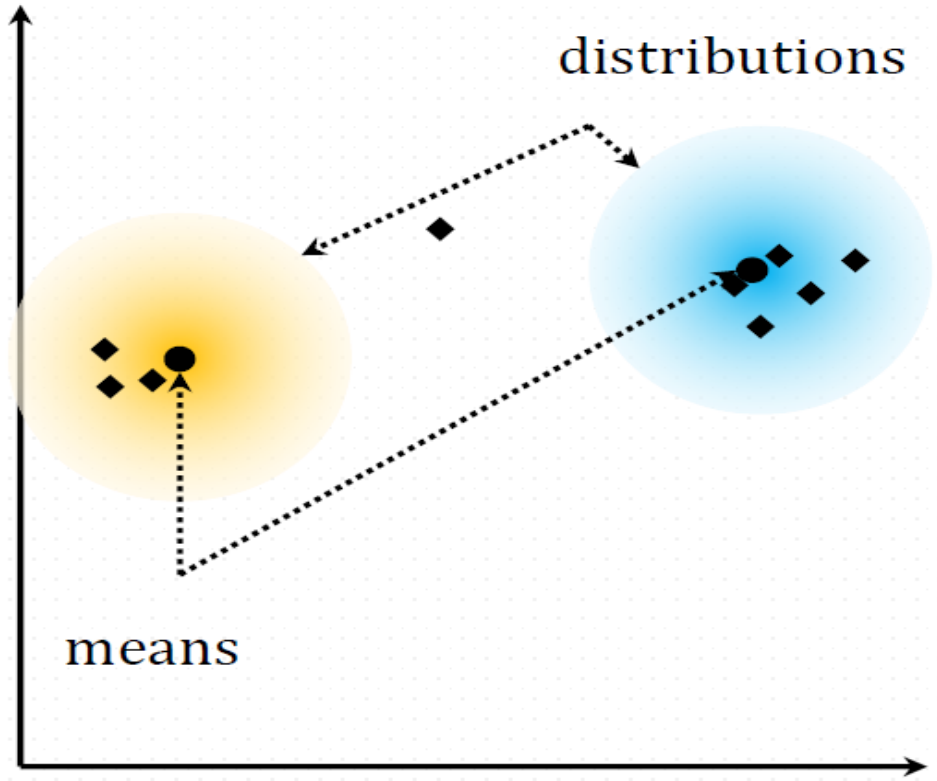
- Compute membership probabilities for each point using bayes rule

EM: Visual Idea: Maximization step



- Recompute means using weighted (the probabilities that a point belong to a distribution) average of points

EM: Visual Idea: Finish



- Repeat expectation and maximization step until no or little change (based on threshold)

Expectation-Maximization (EM)

- Log likelihood with a mixture model

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

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

- 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.

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

$$\text{M-step: } \Phi^{l+1} = \underset{\Phi}{\operatorname{argmax}}\, \mathcal{Q}\left(\Phi \mid \Phi^{l}\right)$$

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 $p(\boldsymbol{x}|G_i) \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$
- E-step:

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

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

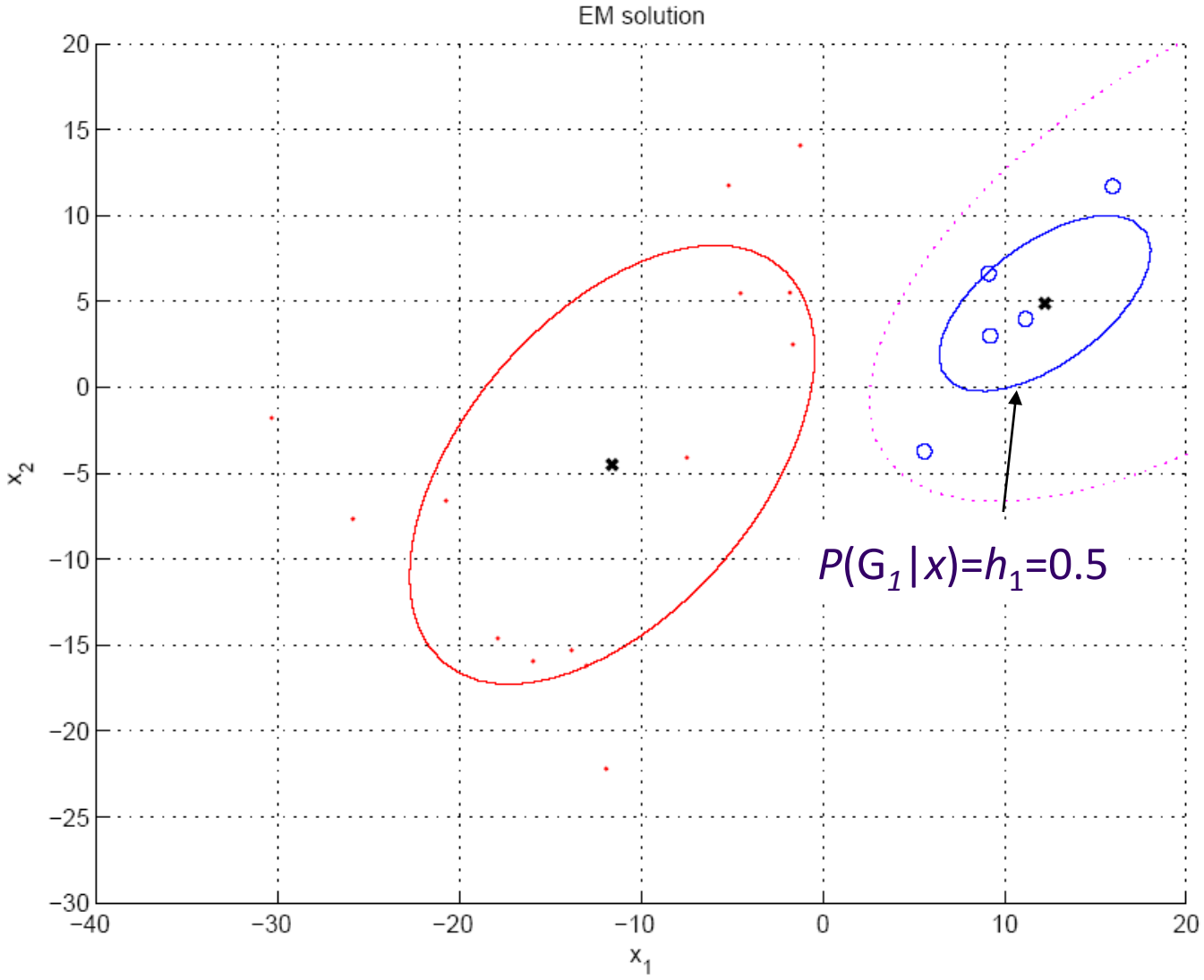
- M-step:

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

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

Use estimated labels in place of unknown labels

EM : Solution


EM solution
20
15
10
5
0
−5
−10
−15
−20
−25
−30
−40
−30
−20
−10
0
10
20
x2
x1
P(G1|x)=h1=0.5

Mixture model clustering using EM: Pros and Cons

- Cons
  - Slow
  - Does not work with few data points
  - Does not work with collinear points
  - Challenging to choose number of clusters
  - Challenging to choose distributions

- Pros
  - Can use distribution of various types
  - Easy Characterization of clusters

Hierarchical Clustering

- Set of nested clusters organized as a tree
- 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

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

Hierarchical clustering

- Does not assume value of k
- Could be more meaningful than K-Mean.

Agglomerative vs Decisive clustering

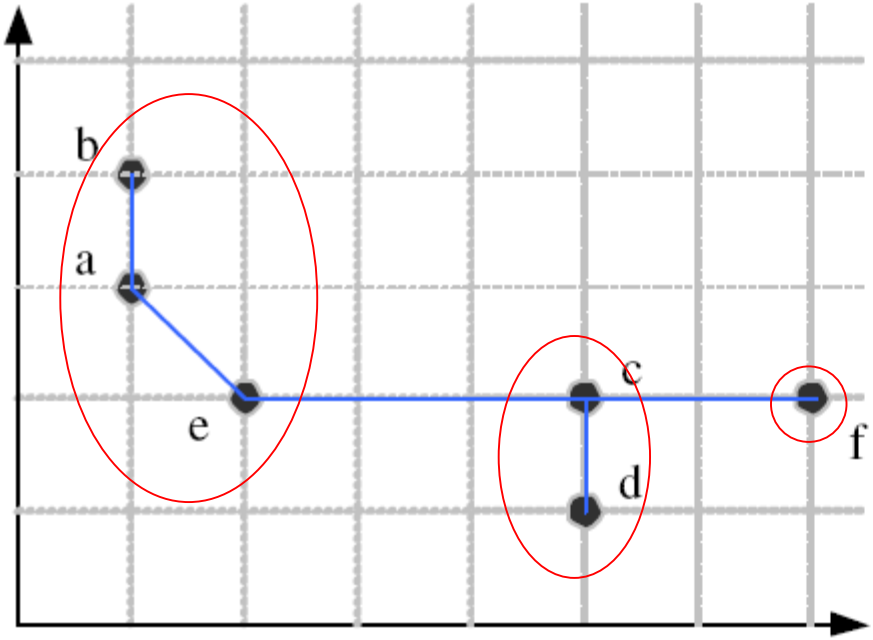
- Agglomerative (Bottom up):
  - Start with N group, merge them based on similarity
- Decisive (Top Down):
  - Start with one group, divide them into smaller group until each group has a single instance.

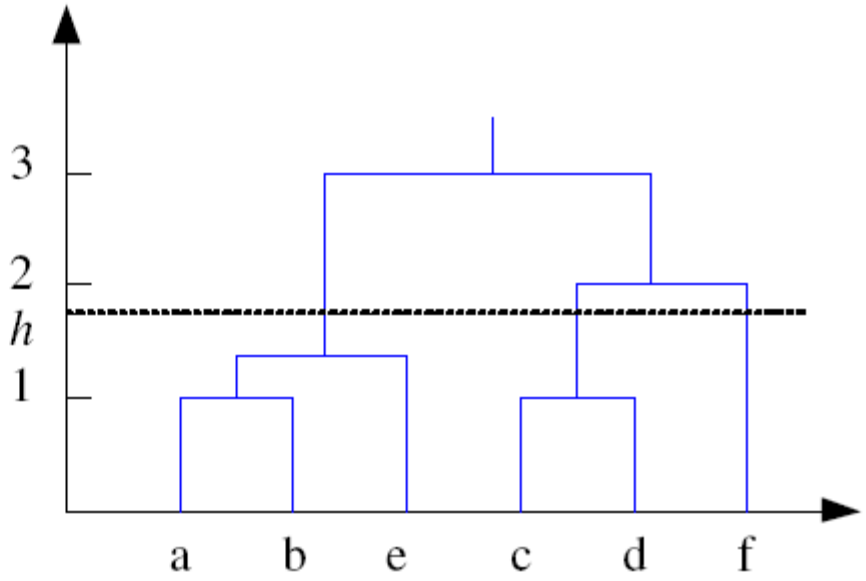
Agglomerative Clustering: details

- Start with *N* groups each with one instance and merge two closest groups at each iteration

- Distance between two groups G_i and G_j:
  - Single-link:
$$d(G_i, G_j) = \min_{\mathbf{x}^r \in G_i, \mathbf{x}^s \in G_j} d(\mathbf{x}^r, \mathbf{x}^s)$$
  - Complete-link:
$$d(G_i, G_j) = \max_{\mathbf{x}^r \in G_i, \mathbf{x}^s \in G_j} d(\mathbf{x}^r, \mathbf{x}^s)$$
  - Average-link: average of distances between all pairs
  - Centroid: distance between the centroids

Example: Single-Link Clustering





Dendrogram

Evaluation of clustering algorithm

- Unsupervised
  - Without any external information.
- Supervised
  - compare with external structure.
- Relative
  - Compares different clusterings or

Cluster Validity

$$overall\ validity = \sum_{i=1}^{K} w_i\ validity(C_i)$$

- *Validity*(C_i)
  - Cohesion: sum of the proximities with respect to prototype (mean/median etc.)
  - Separation: relative proximity of clusters

The Silhouette Coefficient

- For i[th] object $s_i = (b_i - a_i)/\max(a_i, b_i)$
  - a_i: average distance to all other objects in its cluster
  - b_i: minimum of object's average distances to all the objects in a different cluster
- Combines cohesion and separation
- Possible value can be between -1 and 1
- A positive value is desired with $a_i < b_i$ and a_i close to 0.

How to Choose *k*

- Defined by the application
- 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
- Rule of thumb k = sqrt(n/2), not very reliable