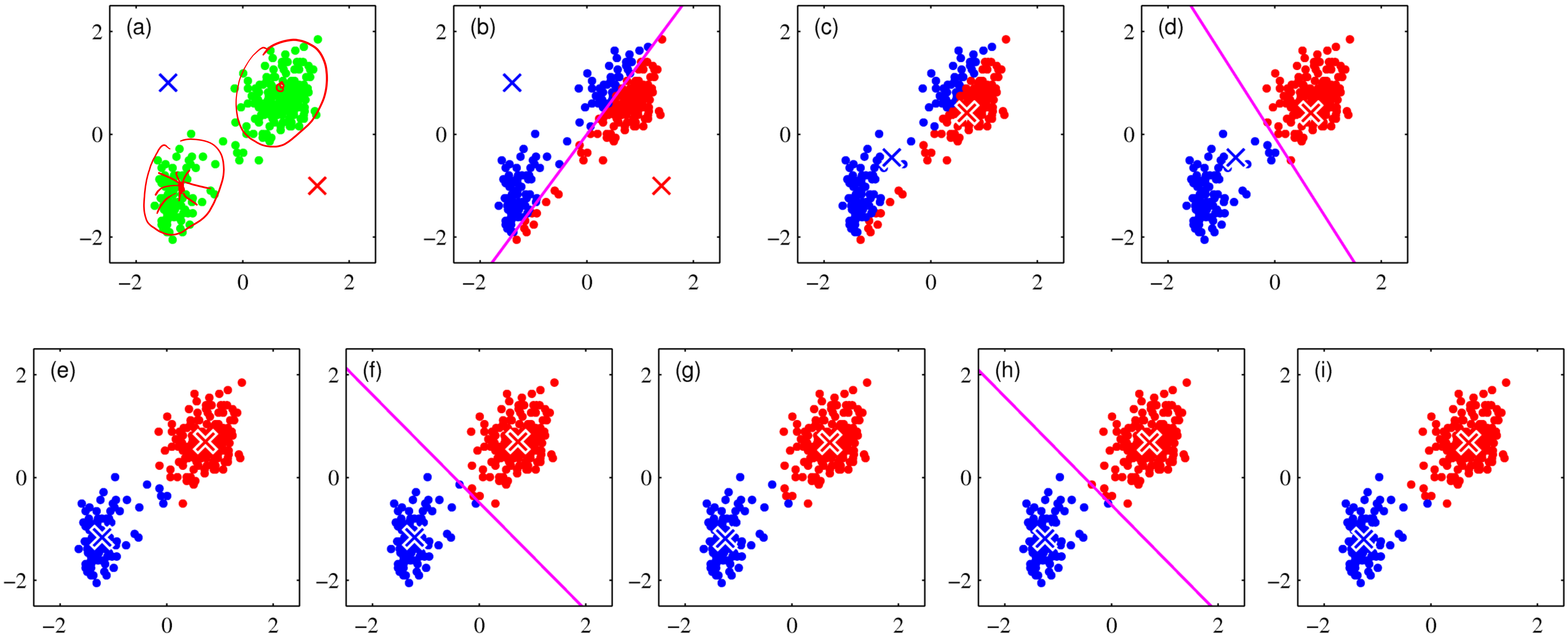
Data Analytics
EEE 4774 & 6777

Module 3

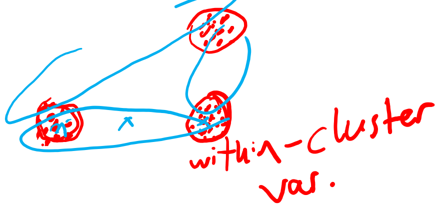
Clustering

Spring 2022

Clustering: K-means



K-means



(intra-cluster)

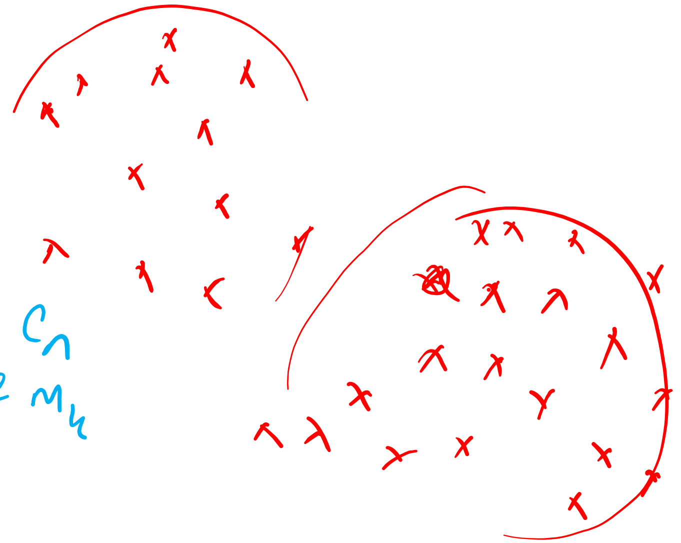
Objective: Minimize the within-cluster variances

- Unsupervised method for identifying groups: Clustering
- Data $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ where $\boldsymbol{x}_n \in \mathbb{R}^D$ #dim.

  #instances #clusters
- min $E(\boldsymbol{c}_n, \boldsymbol{m}_k) \underset{\text{min}}{=} \sum_{n=1}^{N} \sum_{k=1}^{K} c_{nk} \|\boldsymbol{x}_n - \boldsymbol{m}_k\|^2$ where $\boxed{\boldsymbol{c}_n = [c_{n1} \ldots c_{nK}] \text{ and } c_{nk} \in \{0,1\}}$

  cluster mean
- Iteratively minimize E over $\boldsymbol{c}_n$ and $\boldsymbol{m}_k$

  Cluster assignment var.



Initialize $\boldsymbol{m}_k$
for i=1:max_iter
Step1 : Minimize E with respect to $\boldsymbol{c}_n$ keeping $\boldsymbol{m}_k$ fixed → Update c_n
Step2 : Minimize E with respect to $\boldsymbol{m}_k$ keeping $\boldsymbol{c}_n$ fixed → Update m_k
if $\frac{\left\|\boldsymbol{c}_n^{(i)} - \boldsymbol{c}_n^{(i-1)}\right\|}{\left\|\boldsymbol{c}_n^{(i-1)}\right\|} < \varepsilon$ and $\frac{\left\|\boldsymbol{m}_k^{(i)} - \boldsymbol{m}_k^{(i-1)}\right\|}{\left\|\boldsymbol{m}_k^{(i-1)}\right\|} < \varepsilon$
norm
break
end
end

K-means

Step 1:

n^{th} data instance

cluster center

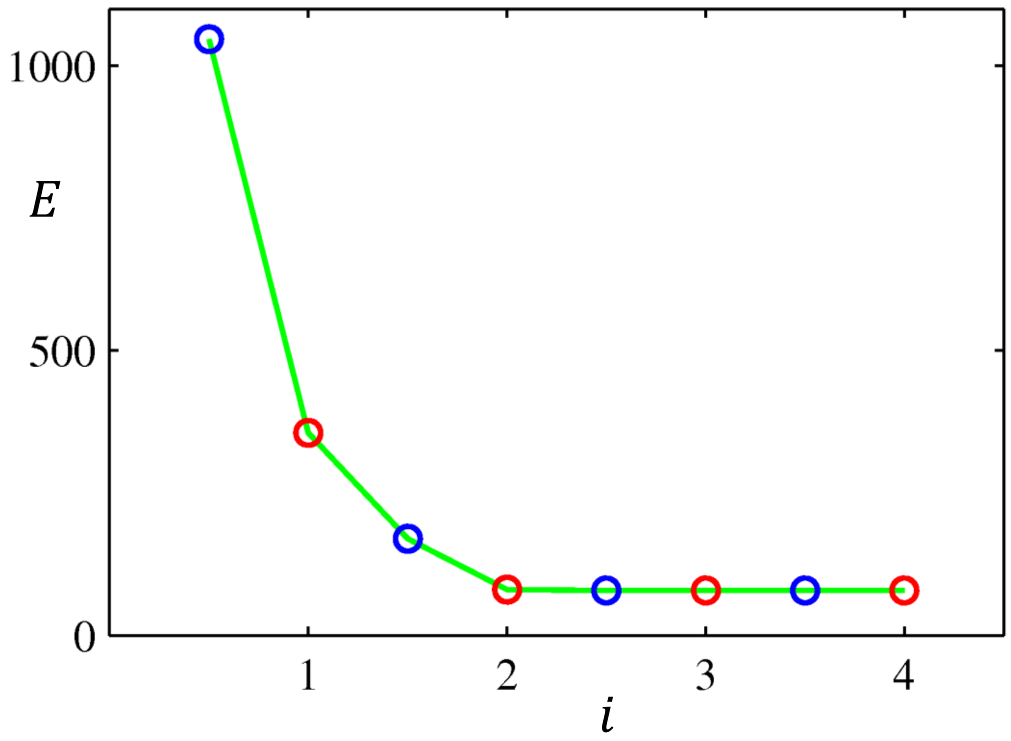
$$c_{nk} = \begin{cases} 1 & if \quad k = \arg\min_j \left\| \boldsymbol{x}_n - \boldsymbol{m}_j \right\|^2 \\ 0 & otherwise \end{cases}$$

Euclidean distance

Assign each data point to the nearest cluster

Step2

$$\boldsymbol{m}_k = \frac{\sum_n c_{nk}\boldsymbol{x}_n}{\underbrace{\sum_n c_{nk}}_{\#\text{ data points in cluster } k}} = mean\ of\ points\ assigned\ to\ cluster\ k$$



- Since E decreases at each iteration, convergence is guaranteed
- However, it may converge to a local minimum
- K-medoids: generalization of K-means to a general distance measure

$$E(\boldsymbol{c}_n, \boldsymbol{m}_k) = \sum_{n=1}^{N}\sum_{k=1}^{K} c_{nk}\, \underbrace{V(\boldsymbol{x}_n, \boldsymbol{m}_k)}_{\text{non-Euclidean distance}}$$

K-means

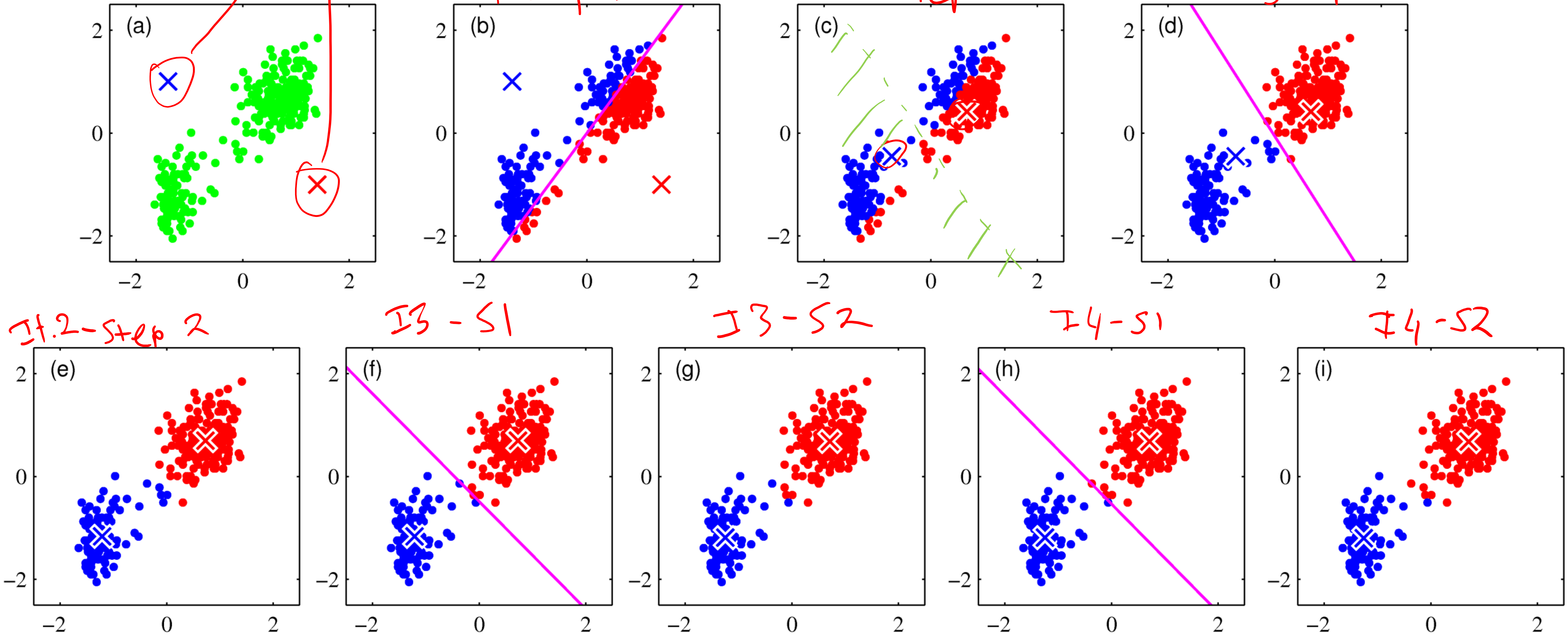
randomly chosen cluster means

Iteration 1

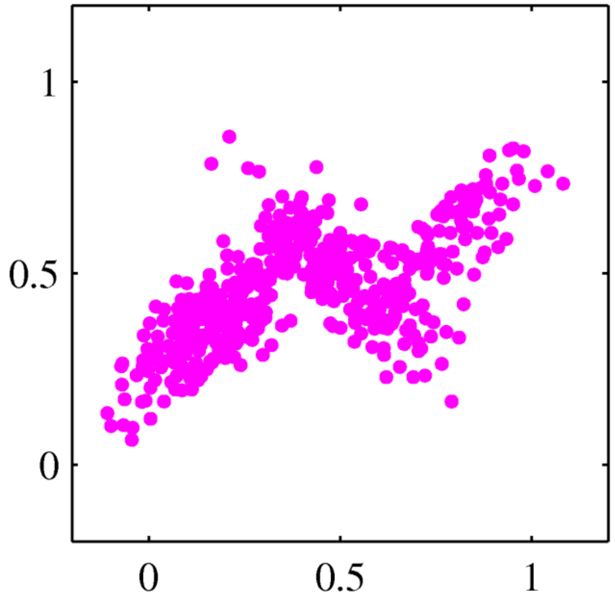
Step 1 Step 2

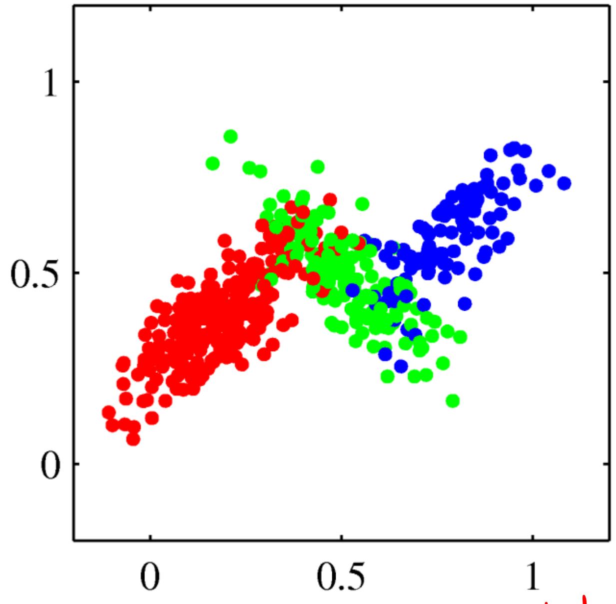
Iter. 2

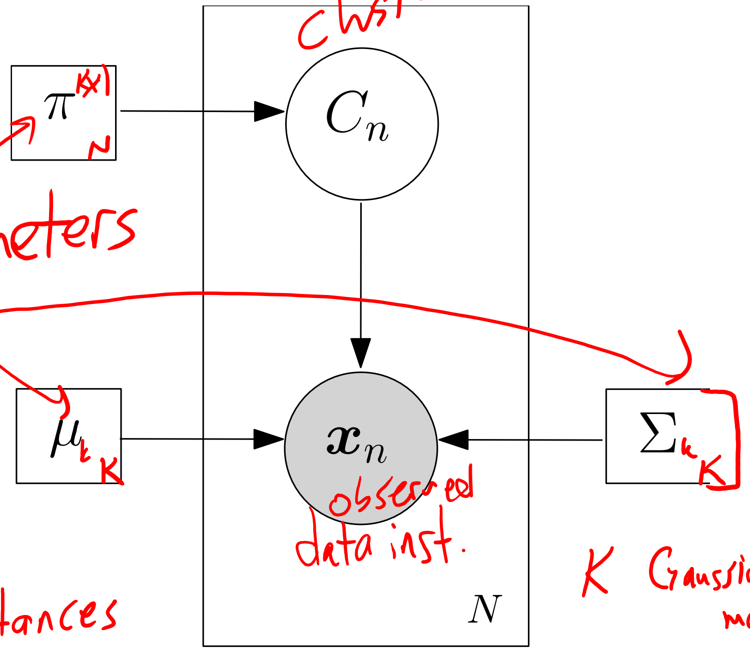
Step 1



Gaussian Mixture Model (GMM)







$$\boldsymbol{C}_n = [C_{nk}]_{k=1,\dots,K} = [0 \cdots 1 \cdots 0] \in [0,1]^K$$

cluster assign/membership var. K clusters

$$p(C_{nk} = 1) = \pi_k, \quad \pi_k \in [0,1], \qquad \sum_{k=1}^{K} \pi_k = 1$$

likelihood of x_n under GMM

latent/hidden var.

categorical distr.

cluster mean

covariance

$$p(\boldsymbol{x}_n) = \sum_{\boldsymbol{C}_n \in [0,1]^K} \underbrace{p(\boldsymbol{x}_n, \boldsymbol{C}_n)}_{\text{joint prob. dist.}} = \sum_{k=1}^{K} \underbrace{\pi_k}_{\text{prior}} \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

likelihood of x_n under k^{th} Gaus. Model

log-likelihood

instances

$$\log p(\boldsymbol{X}) = \sum_{n=1}^{N} \underbrace{\log \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}_{\log p(X_n)}$$

$$\log p(X) = \log \prod_{n=1}^{N} p(x_n) = \sum \log p(x_n)$$

$$= \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k \frac{\exp\{-(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)/2\}}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_k|^{1/2}}$$

ML for GMM

$$\frac{\partial}{\partial \mu_k} \log f(\mu_k) = \frac{\frac{\partial}{\partial \mu_k} f(\mu_k)}{\mu_k}$$

$$\frac{\partial}{\partial \mu_k} e^{f(\mu_k)} = e^{f(\mu_k)} \frac{\partial}{\partial \mu_k} f(\mu_k)$$

prior prob for cluster k

$$\max_{\boldsymbol{\mu}_k} \log p(\boldsymbol{X}) \quad \Longrightarrow \quad \frac{\partial}{\partial \boldsymbol{\mu}_k} \log p(\boldsymbol{X}) = \sum_{n=1}^{N} \frac{\pi_k \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \boldsymbol{\Sigma}_k^{-1} (\boldsymbol{x}_n - \boldsymbol{\mu}_k) = 0$$

derivative of exponent in Gaus. pdf

posterior prob. for cluster k

$$p(C_{nk} = 1 | \boldsymbol{x}_n) = \frac{\overbrace{p(C_{nk} = 1)}^{\pi_k} \overbrace{p(\boldsymbol{x}_n | C_{nk} = 1)}^{\mathcal{N}(x_n | \mu_k, \Sigma_k)}}{\underbrace{\sum_{j=1}^{K} p(C_{nj} = 1)\, p(\boldsymbol{x}_n | C_{nj} = 1)}_{\text{joint prob. } p(x_n, C_n)}} = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} = \gamma(C_{nk})$$

posterior

coupled equations
no closed-form solution!

For $\mu_k, \Sigma_k, \pi_k \Rightarrow$ Need to find $\gamma(C_{nk})$ which depends on μ_k, Σ_k, π_k!

$$\boldsymbol{\mu}_k = \frac{1}{\sum_{n=1}^{N} \gamma(C_{nk})} \sum_{n=1}^{N} \underbrace{\gamma(C_{nk})}_{\text{weight with posterior}} \boldsymbol{x}_n = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(C_{nk})\, \boldsymbol{x}_n$$

weighted average (Sample mean)

as we have multiple gaussian here and we don't know to which gauss. our data point belongs to

Similarly,

$$\boldsymbol{\Sigma}_k = \underbrace{\frac{1}{N_k} \sum_{n=1}^{N} \gamma(C_{nk})\, (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T}_{\text{weighted sample cov. matrix}}, \qquad N_k = \sum_{n=1}^{N} \gamma(C_{nk}),$$

effective number of points in cluster k

and

$$\pi_k = \frac{N_k}{N}$$

as we are doing soft assignment

Iterative Solution: EM for GMM

- Expectation-Maximization for iteratively computing ML in GMM

1. Initialize $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k$ and compute the initial value of $\log p(\boldsymbol{X})$

Iteration

2. **E step:** Compute the posteriors using the current parameter values

$$\gamma(C_{nk}) = \frac{\pi_k \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

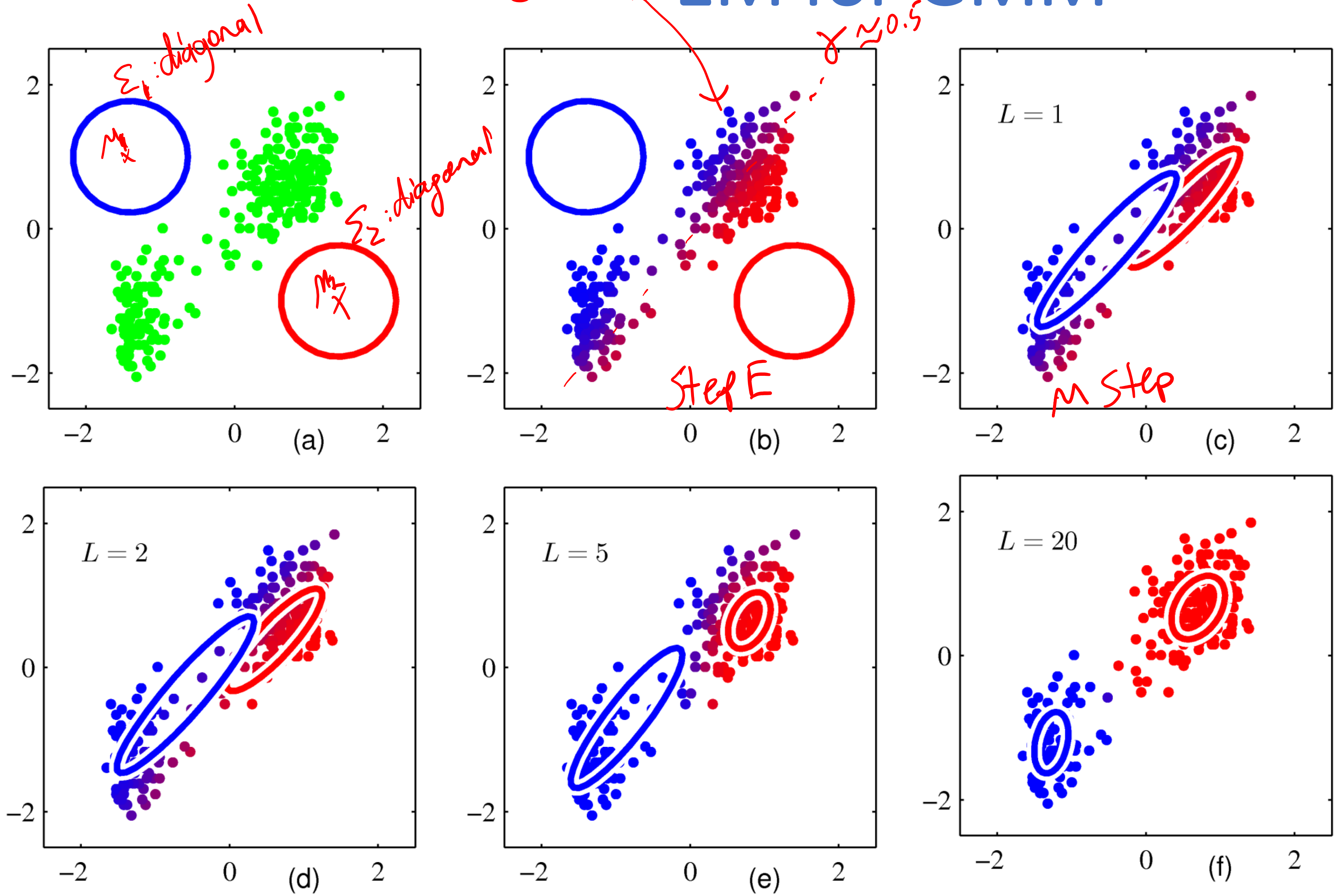
3. **M step:** Re-estimate the parameters using the current posteriors

$$\boldsymbol{\mu}_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(C_{nk})\, \boldsymbol{x}_n\,, \quad \boldsymbol{\Sigma}_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(C_{nk})\, (\boldsymbol{x}_n - \boldsymbol{\mu}_k^{new})(\boldsymbol{x}_n - \boldsymbol{\mu}_k^{new})^T\,, \quad \pi_k^{new} = \frac{N_k}{N}, \quad \text{where} \quad N_k = \sum_{n=1}^{N} \gamma(C_{nk})$$

4. Compute the log-likelihood and check for convergence of either the parameters or the log-likelihood.

   If no convergence, return to step 2.

EM for GMM



- Many more iterations than K-means, and each iteration much more expensive,
- But provides ***probabilistic modeling*** with ***soft assignments*** and ***covariance***
- Run K-means to initialize EM for GMM
- Converges to a local maximum

Expectation-Maximization (EM) Algorithm

used for solving inter-locked equations of latent variable models

- Objective: find ML for models with latent variables $\boldsymbol{C}$ (e.g., missing values in the dataset), observed data $\boldsymbol{X}$, and parameters $\boldsymbol{\theta}$

$$\log p(\boldsymbol{X}|\boldsymbol{\theta}) = \log \sum_{\boldsymbol{C}} p(\boldsymbol{X}, \boldsymbol{C}|\boldsymbol{\theta})$$

- Assume maximization of the complete-data log-likelihood $\log p(\boldsymbol{X}, \boldsymbol{C}|\boldsymbol{\theta})$ is easy

1. Initialize $\boldsymbol{\theta}^{old}$

posterior of latent var. C given data

2. E step: Evaluate $p(\boldsymbol{C}|\boldsymbol{X}, \boldsymbol{\theta}^{old})$ and

complete data

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = E_{\underbrace{p(\boldsymbol{C}|\boldsymbol{X}, \boldsymbol{\theta}^{old})}_{\text{posterior}}} [\underbrace{\log p(\boldsymbol{X}, \boldsymbol{C}|\boldsymbol{\theta})}_{\text{complete data log-likelihood}}] = \sum_{\boldsymbol{C}} p(\boldsymbol{C}|\boldsymbol{X}, \boldsymbol{\theta}^{old}) \log p(\boldsymbol{X}, \boldsymbol{C}|\boldsymbol{\theta})$$

3. M step: $\boldsymbol{\theta}^{new} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$ {maximize $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) + \underbrace{\log p(\boldsymbol{\theta})}_{\text{prior for param. } \theta}$ for MAP}

4. If no convergence, then $\boldsymbol{\theta}^{old} \leftarrow \boldsymbol{\theta}^{new}$ and return to step 2

GMM by EM vs. K-means

- EM soft assigns data points ***softly*** to a cluster using posterior $p(C_{nk} = 1|\boldsymbol{x}_n)$,

whereas K-means performs ***hard*** assignment

- Consider a GMM with covariance $\epsilon\boldsymbol{I}$ for all clusters, where ϵ is a fixed constant, not a parameter to be re-estimated

$$p(C_{nk} = 1|\boldsymbol{x}_n) = \frac{\pi_k \exp\{-\|\boldsymbol{x}_n - \boldsymbol{\mu}_k\|^2/2\epsilon\}}{\sum_{j=1}^{K} \pi_j \exp\{-\left\|\boldsymbol{x}_n - \boldsymbol{\mu}_j\right\|^2/2\epsilon\}}$$

- As $\epsilon \to 0$, in the denominator the smallest $\left\|\boldsymbol{x}_n - \boldsymbol{\mu}_j\right\|^2$ will go to 0 most slowly,

hence posterior for that cluster will go to 1 and the others will go to 0 $\Longrightarrow$ ***Hard assignment to the closest cluster***

- Update for the mean $\boldsymbol{\mu}_k$ also reduces to that of K-means

- K-means does not estimate the covariances of the clusters

Evaluation of Clustering Results

- Several similarity measures for clusters can be used to evaluate the performance of clustering algorithms
- Can be used to determine the optimum number of clusters
- **Internal Evaluation:** based on the clustered data itself
  - typically assigns good score if high similarity within clusters and low similarity between clusters
  - e.g., Silhouette value (works well with K-means), Dunn index, Davies-Bouldin index
- **External Evaluation:** based on data that was not used for clustering, e.g., ground truth
  - measures how close clustering is to the benchmark classes
  - e.g., Rand index, F-measure, Mutual information, Confusion matrix

    adjusted Rand index