Mathematics for Machine Learning (AI 512):

Module 4

Amit Chattopadhyay

IIIT-Bangalore

Module 4:

- **K-Means**: Hard and Soft Clustering
- **Gaussian Mixture Model (GMM)**
- **Expectation-Maximization** (EM) framework
- **Expectation-Maximization** for **GMM** parameter estimation

Reference books:

1. *Mathematics for Machine Learning*, by Marc Peter Deisenroth, A. Aldo Faisal and Cheng Soon Ong. (Ch 5, **Ch. 11**)
2. *Pattern Recognition and Machine Learning*, by Chrisopher M. Bishop. (Ch 2.3, **Ch. 9**)
3. *Machine Learning - A Probabilistic Perspective* by Kevin Murphy, Ch 11

K-Means Clustering

Problem:

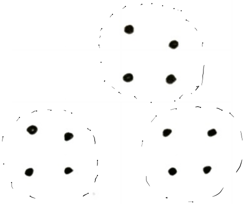
How to cluster a given a set $\mathbf{X} = \{\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N\}$ of **unsupervised** or **unlabeled** data?

- $$\underline{x}_i = \begin{bmatrix} x_{i_1} \\ x_{i_2} \\ \vdots \\ x_{i_d} \end{bmatrix} = (x_{i_1}, x_{i_2}, \ldots, x_{i_d}) \in \mathbb{R}^d: \ d\text{-dimensional feature vector}$$

  (for $i = 1, 2, \ldots, N$)

K=3

- **Number of clusters:** Assumed K to be known. **Cluster validation** (not part of current discussion).



K-Means Clustering

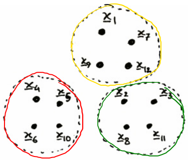
- **X**: unlabeled data,
  K: number of clusters (known)

Data:	$\underline{x}_1$	$\underline{x}_2$	$\cdots$	$\underline{x}_n$	$\cdots$	$\underline{x}_N$
Class-Label:	y_1	y_2	$\cdots$	y_n	$\cdots$	y_N

- If $y_n \in \{1, 2, \ldots, K\}$ (for all n) are determined, the cluster is realized
- **Clustering Problem:** What is the **optimal** labeling?

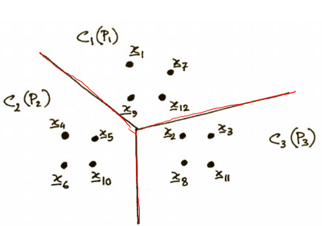
Criteria:

1. High **intra-class** similarity
2. Low **inter-class** similarity

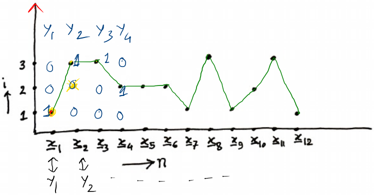


K-Means Clustering: Trellis View

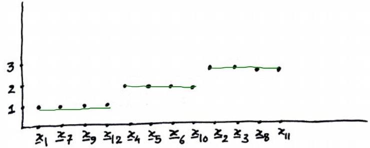
$\mathbb{R}^d$



$\gamma_{nk} = P(y_n = k \mid \underline{x}_n) = 1$



total 3^N curves



K-Means Clustering and Latent Variables

- **Latent variables:** hidden inside the data, e.g. $\mathbf{Y} = \{y_1, \ldots, y_N\}$. Once known the data information $\{\mathbf{X}, \mathbf{Y}\}$ is complete.
- If complete data $\{\mathbf{X}, \mathbf{Y}\}$ is known, parameter estimation using maximization of complete-data log likelihood: $\ln p(\mathbf{X}, \mathbf{Y};\ \theta)$ is straight-forward.
- In practice, complete data is not known, but have only the incomplete data $\mathbf{X}$.

Initialize: K **cluster centroids**:
$\{\underline{\mu}_1, \underline{\mu}_2, \ldots, \underline{\mu}_K\}$

D D'

See the logic how D is minimized at every iteration.

To compute parameters:
$\underline{\theta}^* = (\underline{\mu}_1^*, \underline{\mu}_2^*, \ldots, \underline{\mu}_K^*)$ such that

$$\underline{\theta}^* = \underset{\underline{\theta}}{\text{argmin}}\ D(\mathbf{X}, \underline{\theta})$$

The same logic will imply D' is maximized.

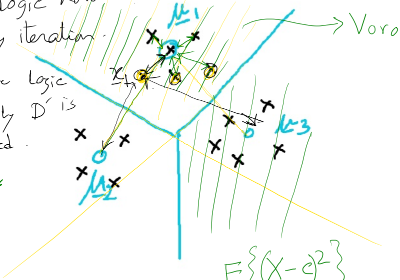
$D'(\underline{x}, \underline{\theta}) =$

where $\boxed{D(\mathbf{X}, \underline{\theta}) = \sum_{k=1}^{K} \boxed{\sum_{\underline{x}_n \in C_k} d(\underline{x}_n, \underline{\mu}_k)}}$

In particular, $d(\underline{x}_n, \underline{\mu}_k) = \|\underline{x}_n - \underline{\mu}_k\|^2$.

Voronoi Diagram

→ Voronoi Cell/Region



$E\{(X-c)^2\}$
$= E\{(X-\mu+\mu-c)^2\}$
$= E\{(X-\mu)^2 + \underline{(\mu-c)^2} + 2(X-\mu)(\mu-c)\}$
$= E((X-\mu)^2) + (\mu-c)^2 + 2(\mu-c)E(X-\mu)$

Q1: Why labeling of points is equivalent to finding parameters $\underline{\mu}_i$'s?
Q2: How *intra-class similarity* is maximized in the above formula?
Q3: How *inter-class similarity* is minimized in the above formula?

K-Means Clustering vs. Parameter Estimation in GMM

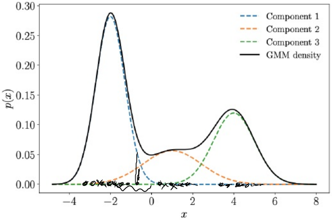
- **Gaussian Mixture Model (GMM):** Density estimation by weighted sum of Gaussians
- GMM: Density model to combine K Gaussian distributions:

$\Sigma_k = (\cdot)_{d \times d}$

$$\boxed{p(\underline{x};\ \underline{\theta}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\underline{x};\ \underline{\mu}_k, \Sigma_k)}$$

$K + dK + \frac{d(d+1)K}{2}$

where $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^{K} \pi_k = 1$. $\underline{\theta} = \{\underline{\mu}_k, \Sigma_k, \pi_k : k = 1, 2, \ldots, K\}$.



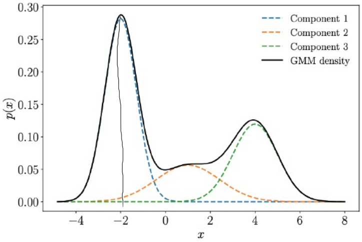
K-Means Clustering vs. Parameter Estimation in GMM

- Finding $\underline{\theta}^*$ in GMM:

$$\hat{\theta}^* = \underset{\underline{\theta}}{\text{argmax}} \prod_{n=1}^{N} p(\underline{x}_n; \underline{\theta})$$

- Finding $\underline{\theta}^*$ in K-Means: *is a special case of finding GMM parameters*

(HW)



Q: How to show K-Means and GMM methods are equivalent?

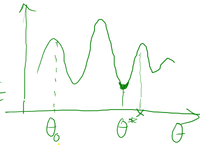
K-Means Clustering: Algorithm

Input: data $\mathbf{X} = \{\underline{x}_1, \ldots, \underline{x}_N\}$ and number of clusters K

$D^{(1)} > D^{(2)} > D^{(3)} > \cdots > D^{*}$

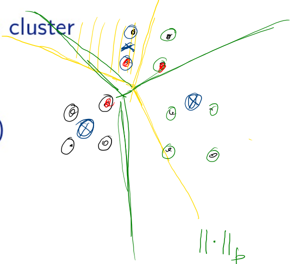
1. **Initialize** cluster centroids: $\underline{\mu}_1^{(0)}, \underline{\mu}_2^{(0)}, \ldots, \underline{\mu}_K^{(0)}$, randomly.
   (**usual trick**: pick K data points)
2. **Repeat** for $t = 1, 2, \ldots$ until convergence
   (till cluster centroids do not change):

   $$\frac{|D^{(t+1)} - D^{(t)}|}{D^{(t+1)}} < \epsilon$$



   (a) For each data point color (label) based on its nearest cluster centroid:

   $$\boxed{y_n^{(t)}} = \operatorname*{argmin}_k \|\underline{x}_n - \underline{\mu}_k^{(t)}\| \quad (n = 1, 2, \ldots, N)$$



   (b) For $k = 1, 2, \ldots, K$

   $$\underline{\mu}_k^{(t+1)} := \frac{\sum_{n=1}^{N} \mathbf{1}\{y_n^{(t)} = k\}\underline{x}_n}{\sum_{n=1}^{N} \mathbf{1}\{y_n^{(t)} = k\}} = \frac{\sum_{n=1}^{N} r_{nk}^{(t)} \underline{x}_n}{\sum_{n=1}^{N} r_{nk}^{(t)}}$$

$\|\cdot\|_p$

$$\operatorname*{argmin}_c \sum_{x_n \in D^{(t)}} \|x_n - c\|_p$$

K-Means Clustering: Algorithm

$$r_{nk}^{(t)} = \begin{cases} 1 & \text{if } x_n \in C_k \\ 0 & \text{if } x_n \notin C_k \end{cases}$$

- Hard Clustering
- Soft Clustering: posterior probabilities

K-Means Clustering: Demo

Show K-Means Demo:

K-Means Clustering: Convergence

Step 2(a): Given centroids $\mu_i^{(t)}$ $(i = 1, 2, \ldots K)$

Find $P_i^{(t)} =$ Optimal NN partition of $\mu_i^{(t)}$

- Each x_n in $P_i^{(t)}$ has a "**closer**" centroid than in $P_i^{(t-1)}$
- $D^{(t)} = \sum_{k=1}^{K} \sum_{x_n \in P_i^{(t-1)}} d(\underline{x}_n, \underline{\mu}_k^{(t)})$ and $D'^{(t)} = \sum_{k=1}^{K} \sum_{x_n \in P_i^{(t)}} d(\underline{x}_n, \underline{\mu}_k^{(t)})$ then $D'^{(t)} \leq D^{(t)}$.

K-Means Clustering: Convergence

<u>Step 2(b)</u>: Given $P_i^{(t)}$

Find $\mu_i^{(t+1)}$ = optimal centroid of data points in $P_i^{(t)}$.

- $d(\underline{x}_n, \underline{\mu}_k^{(t+1)}) \leq d(\underline{x}_n, \underline{\mu}_k^{(t)})$.
  i.e. each x_n has a "**closer**" centroid than $\mu_i^{(t)}$ at $t+1$
- $D'^{(t)} \leq D^{(t+1)}$

K-Means Clustering: Convergence

Q: How to show, at each time t, the updates in K-means algorithm minimizes D?

Thus $D^{(t)} \leq D^{(t+1)}$

i.e. : $D^{(1)} > D^{(2)} > \ldots > D^{(t)} > D^{(t+1)} > \ldots > D^{*}$?