CS145: INTRODUCTION TO DATA MINING

10: Vector Data: Mixture Model

Instructor: Yizhou Sun

yzsun@cs.ucla.edu

November 5, 2018

Methods to Learn

	Vector Data	Set Data	Sequence Data	Text Data
Classification	**Logistic Regression; Decision Tree; KNN SVM; NN**			Naïve Bayes for Text
Clustering	**K-means; hierarchical clustering; DBSCAN; Mixture Models**			PLSA
Prediction	**Linear Regression** GLM*			
Frequent Pattern Mining		Apriori; FP growth	GSP; PrefixSpan	
Similarity Search			DTW	

Vector Data: Mixture Model

- Revisit K-means

- Mixture Model and EM algorithm

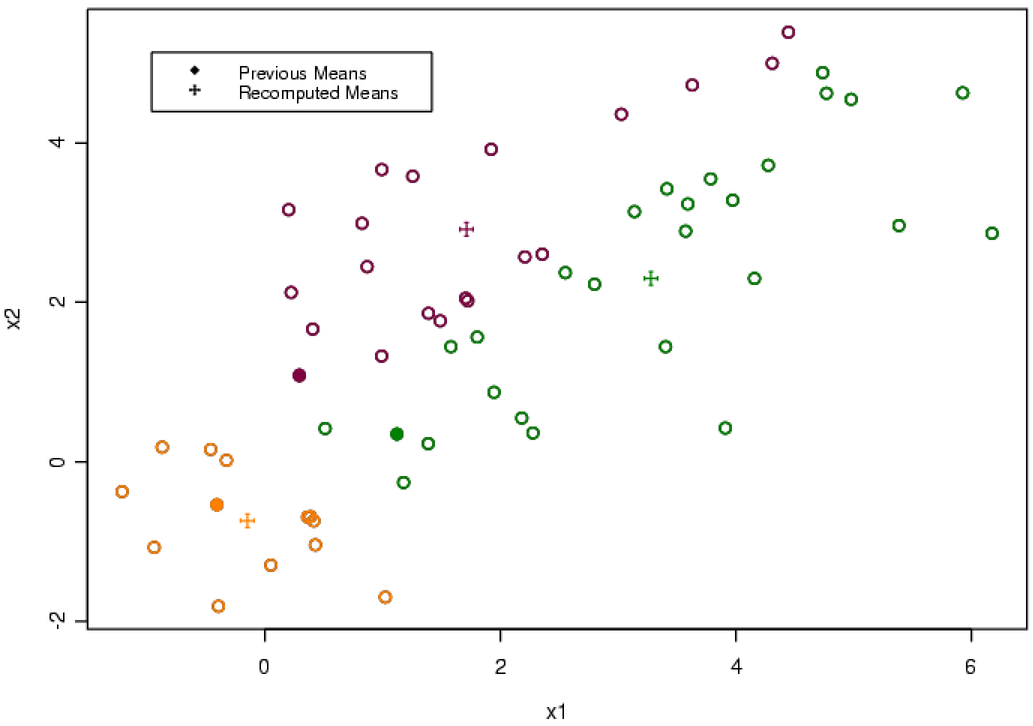
- Summary

Recall K-Means

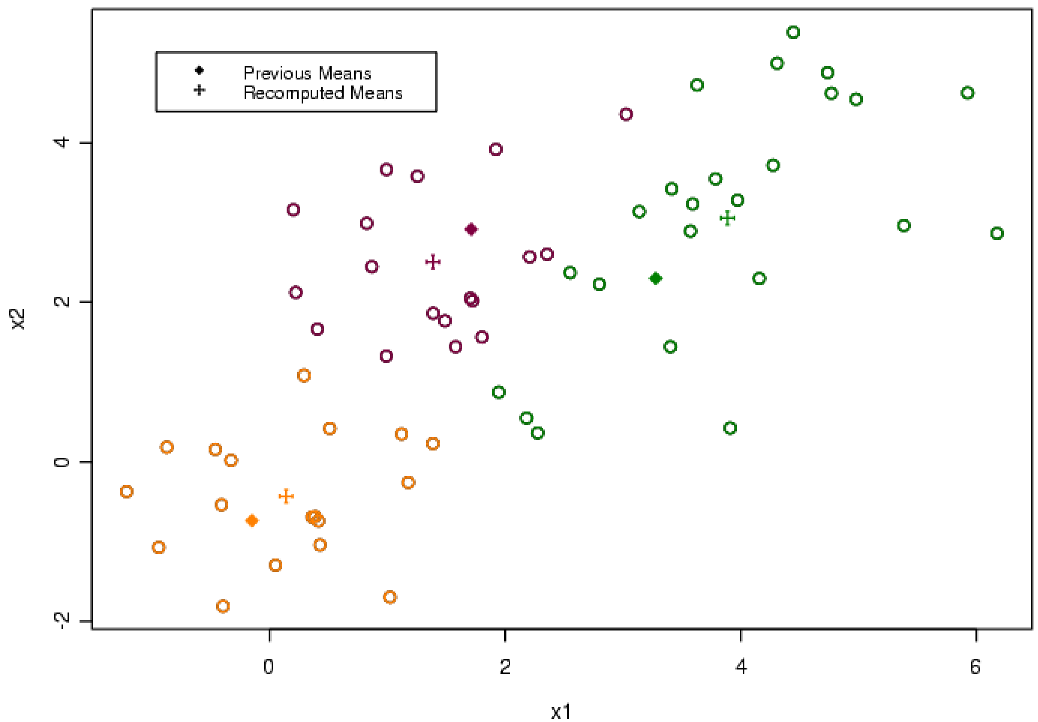
- Objective function
  - $J = \sum_{j=1}^{k} \sum_{C(i)=j} ||x_i - c_j||^2$
  - Total within-cluster variance
- Re-arrange the objective function
  - $J = \sum_{j=1}^{k} \sum_{i} w_{ij} ||x_i - c_j||^2$
    - $w_{ij} \in \{0,1\}$
    - $w_{ij} = 1, if\ x_i\ belongs\ to\ cluster\ j; w_{ij} = 0, otherwise$
  - Looking for:
    - The best assignment w_{ij}
    - The best center c_j

Solution of K-Means

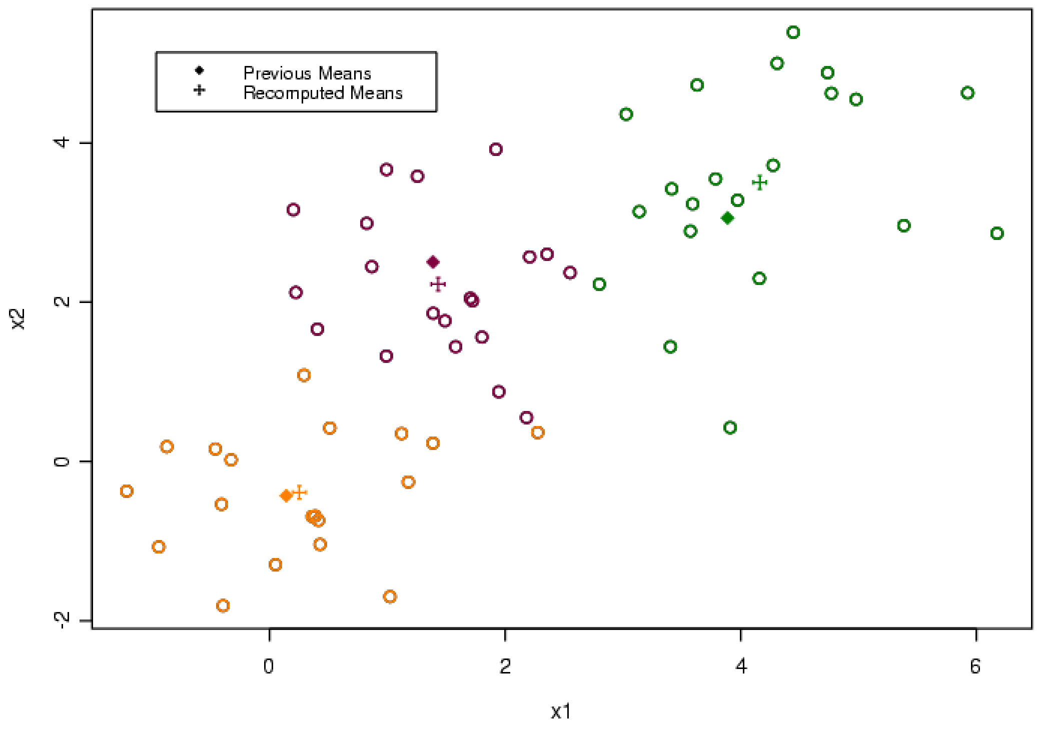
- Iterations $J = \sum_{j=1}^{k} \sum_{i} w_{ij} ||x_i - c_j||^2$
  - Step 1: Fix centers c_j, find assignment w_{ij} that minimizes J
    - => $w_{ij} = 1, if\ ||x_i - c_j||^2$ is the smallest
  - Step 2: Fix assignment w_{ij}, find centers that minimize J
    - => first derivative of J = 0
    - => $\frac{\partial J}{\partial c_j} = -2 \sum_i w_{ij}(x_i - c_j) = 0$
    - =>$c_j = \frac{\sum_i w_{ij} x_i}{\sum_i w_{ij}}$
      - Note $\sum_i w_{ij}$ is the total number of objects in cluster j

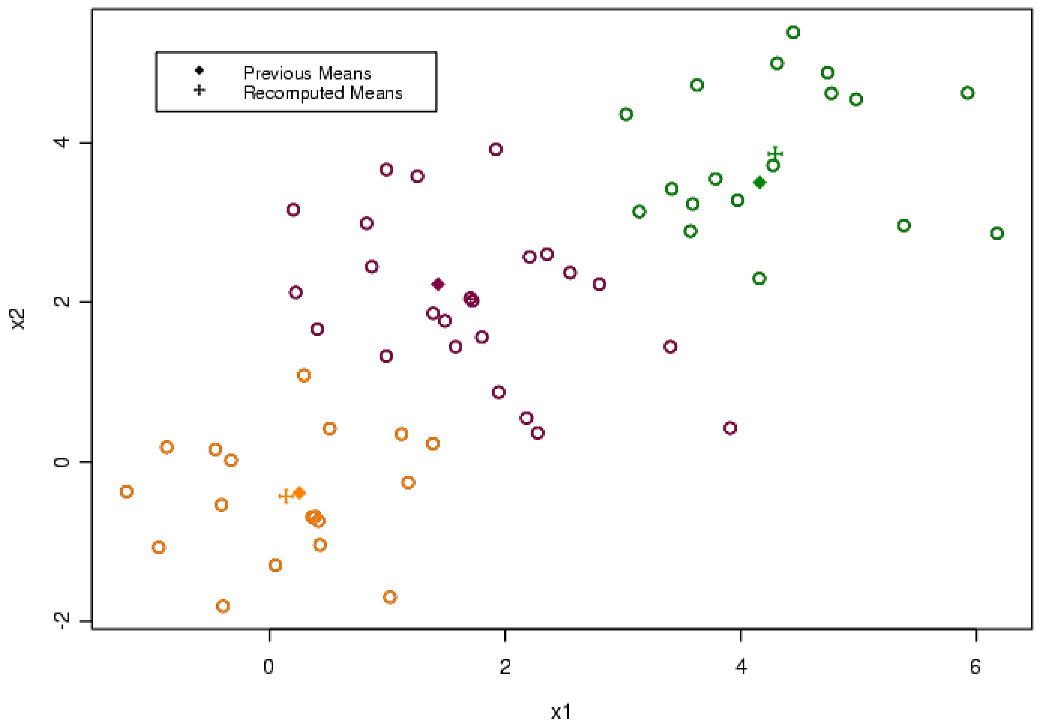


Previous Means
Recomputed Means
x2
x1
-2
0
2
4
0
2
4
6

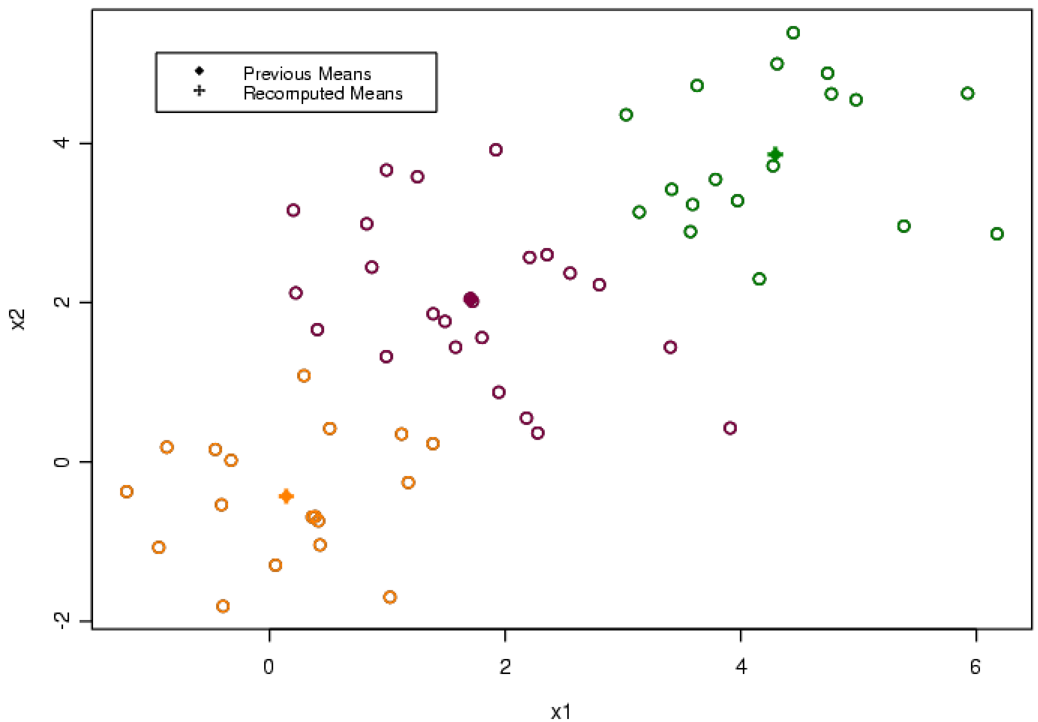


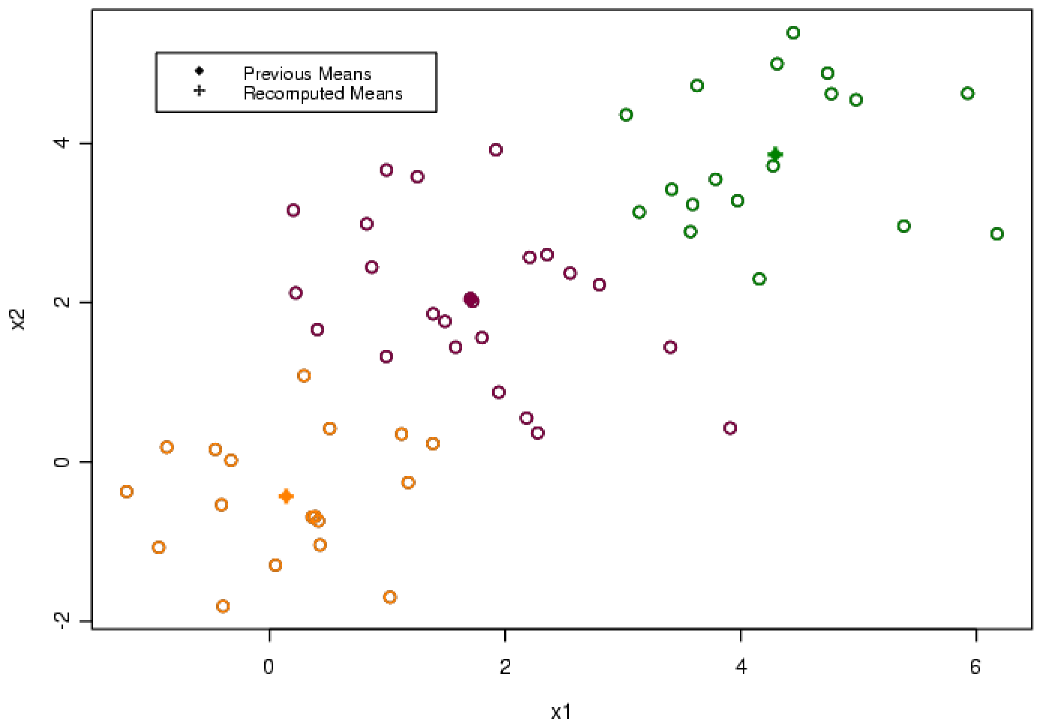
Previous Means
Recomputed Means
x2
x1
-2
0
2
4
0
2
4
6


Previous Means
Recomputed Means
x1
x2
0
2
4
6
-2
0
2
4



Previous Means
Recomputed Means
x2
x1
-2
0
2
4
6


Previous Means
Recomputed Means
x2
x1
-2
0
2
4
6

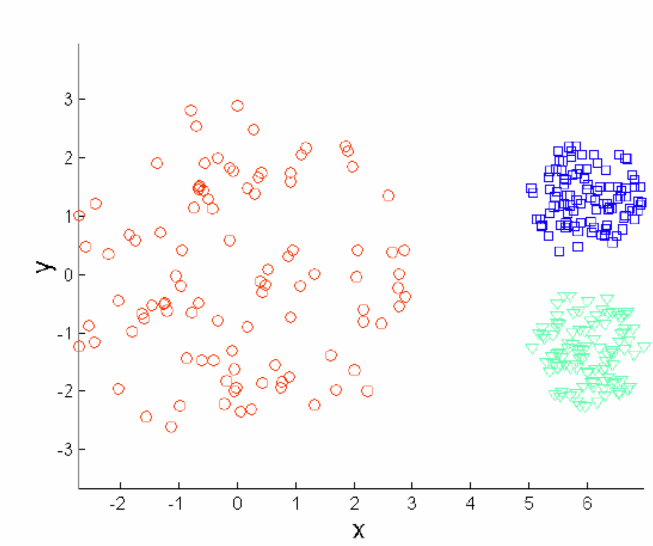


Converges! Why?

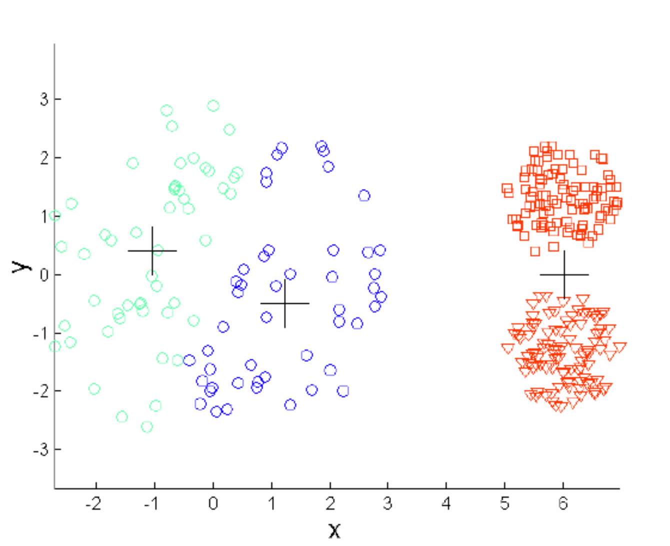
Limitations of K-Means

- K-means has problems when clusters are of different
  - Sizes and density
  - Non-Spherical Shapes

Limitations of K-Means: Different Sizes and Variances



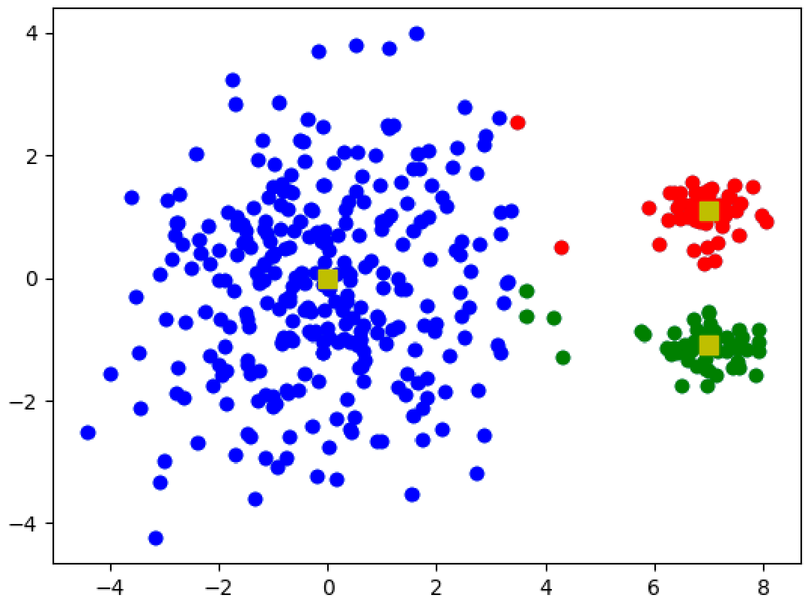
Original Points



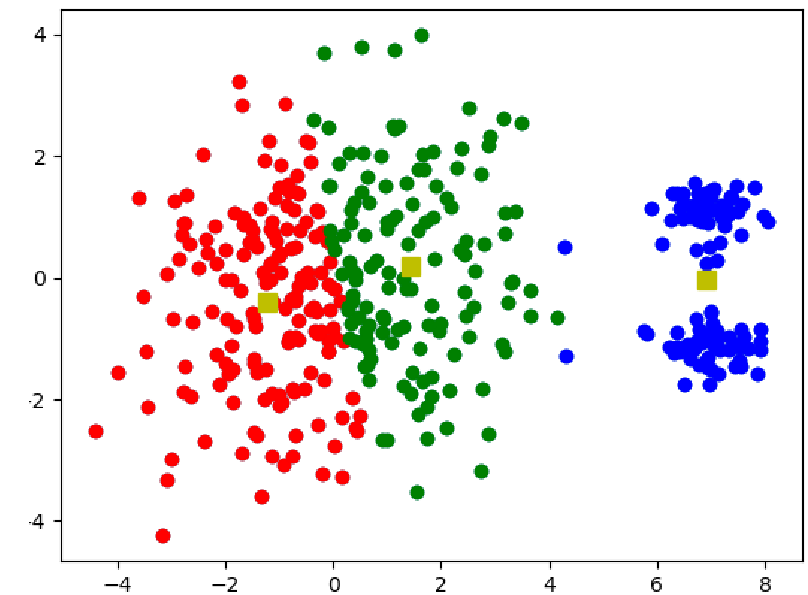
K-means (3 Clusters)

Example

- Consider the cost of K-means in two cases



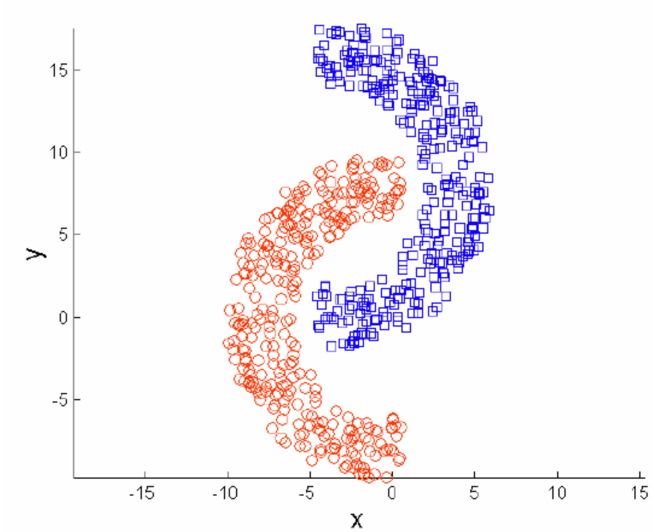
Cost: J = 1560.86



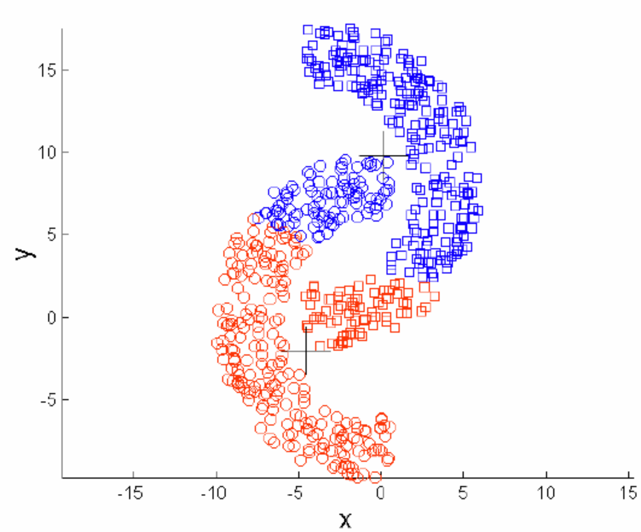
Cost: J = 1147.42

Recall: $J = \sum_{j=1}^{k} \sum_{C(i)=j} ||x_i - c_j||^2$

Limitations of K-Means: Non-Spherical Shapes



Original Points



K-means (2 Clusters)

Vector Data: Mixture Model

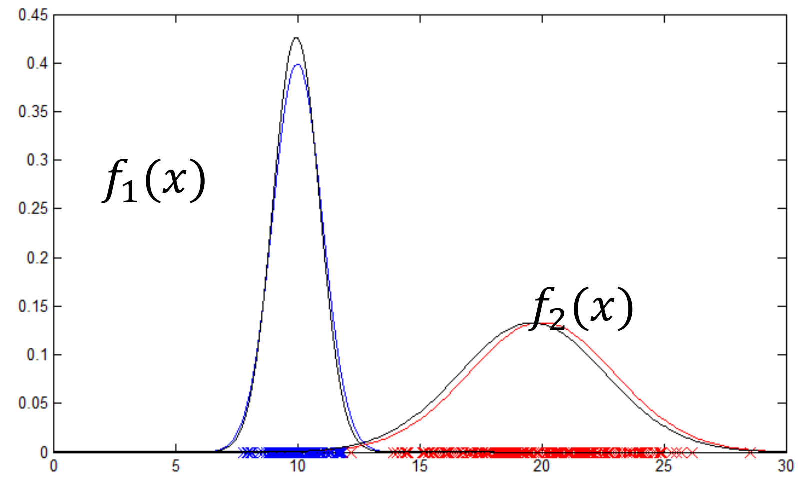
- Revisit K-means
- Mixture Model and EM algorithm
- Summary

Hard Clustering vs. Soft Clustering

- Hard Clustering
  - Every object *i* is assigned to one cluster *j*, e.g., k-means
    - $w_{ij} = \{0,1\}\ and\ \sum_j w_{ij} = 1$
- Soft Clustering
  - Every object *i* is assigned with a probability to different clusters
    - $w_{ij} \in [0,1]\ and\ \sum_j w_{ij} = 1$

Mixture Model-Based Clustering

- A set C of k probabilistic clusters $C_1, \ldots, C_k$
  - probability density functions: $f_1, \ldots, f_k$,
  - Cluster prior probabilities: $w_1, \ldots, w_k$, $\sum_j w_j = 1$
- Joint Probability of an object *i and its* cluster C_j is:
  - $p(x_i, z_i = C_j) = w_j f_j(x_i)$
  - z_i: hidden random variable
- Probability of *i* is:
  - $p(x_i) = \sum_j w_j f_j(x_i)$



Maximum Likelihood Estimation

- Since objects are assumed to be generated independently, for a data set D = $\{x_1, \ldots, x_n\}$, we have,

$$p(D) = \prod_i p(x_i) = \prod_i \sum_j w_j f_j(x_i)$$

$$\Rightarrow log p(D) = \sum_i log p(x_i) = \sum_i log \sum_j w_j f_j(x_i)$$

- Task: Find *k* probabilistic clusters s.t. *p*(*D*) is maximized

The EM (Expectation Maximization) Algorithm

- **The (EM) algorithm:** A framework to approach maximum likelihood or maximum a posteriori estimates of parameters in statistical models.
  - **E-step** assigns objects to clusters according to the current soft clustering or parameters of probabilistic clusters

    $f_j(x_i)$ w_j

    - $w_{ij}^{t+1} = p\left(z_i = j \middle| \theta_j^t, x_i\right) \propto p\left(x_i \middle| z_i = j, \theta_j^t\right) p(z_i = j)$
  - **M-step** finds the new clustering or parameters that maximize the expected likelihood, with respect to conditional distribution $p\left(z_i = j \middle| \theta_j^t, x_i\right)$
    - $\theta^{t+1} = argmax_\theta \sum_i \sum_j w_{ij}^{t+1} \log L(x_i, z_i = j|\theta)$

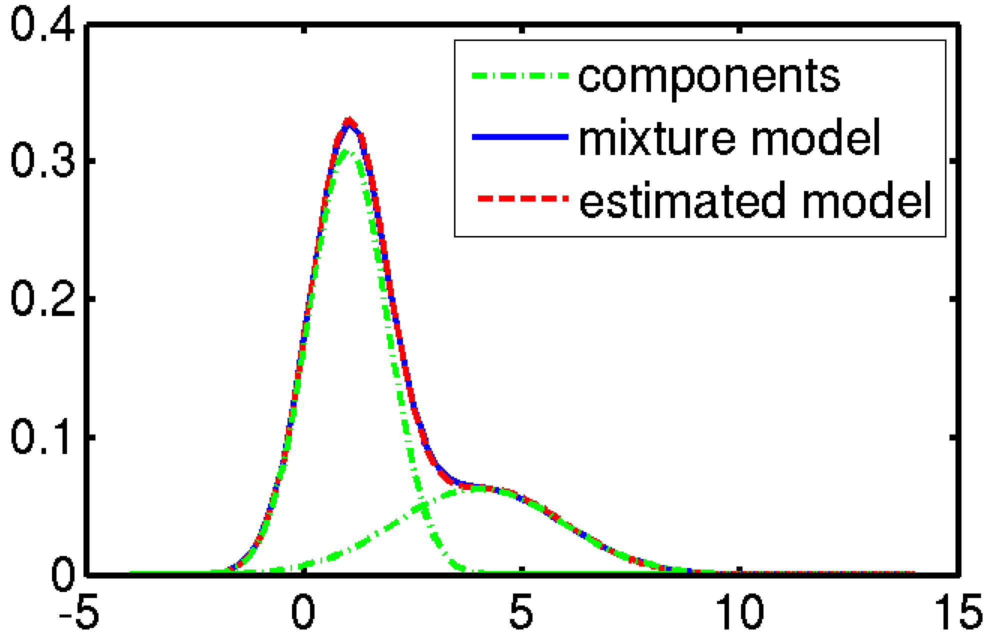
Gaussian Mixture Model

- Generative model
  - For each object:
    - Pick its cluster, i.e., a distribution component: $Z \sim Multinoulli(w_1, \ldots, w_k)$
    - Sample a value from the selected distribution: $X|Z \sim N(\mu_Z, \sigma_Z^2)$
- Overall likelihood function
  - $L(D|\,\theta) = \prod_i \sum_j w_j p(x_i|\mu_j, \sigma_j^2)$
    s.t. $\sum_j w_j = 1 \; and \; w_j \geq 0$
  - Q: What is θ here?

Apply EM algorithm: 1-d

- An iterative algorithm (at iteration t+1)
  - E(expectation)-step
    - Evaluate the weight w_{ij} when μ_j, σ_j, w_j are given
    - $w_{ij}^{t+1} = \frac{w_j^t p(x_i|\mu_j^t,(\sigma_j^2)^t)}{\sum_k w_k^t p(x_i|\mu_k^t,(\sigma_k^2)^t)}$ — $p(x_i|\mu_j^t,(\sigma_j^2)^t) = f_j^t(x_i)$
  - M(maximization)-step
    - Find μ_j, σ_j, w_j that maximize the weighted log likelihood, where w_{ij}'s are the weights: $\sum_{ij} w_{ij}^{t+1} log w_j p(x_i|\mu_j, \sigma_j^2)$
    - It is equivalent to Gaussian distribution parameter estimation when each point has a weight belonging to each distribution
    - $\mu_j^{t+1} = \frac{\sum_i w_{ij}^{t+1} x_i}{\sum_i w_{ij}^{t+1}}$; $(\sigma_j^2)^{t+1} = \frac{\sum_i w_{ij}^{t+1}\left(x_i - \mu_j^{t+1}\right)^2}{\sum_i w_{ij}^{t+1}}$; $w_j^{t+1} = \sum_i w_{ij}^{t+1} / n$

Example: 1-D GMM

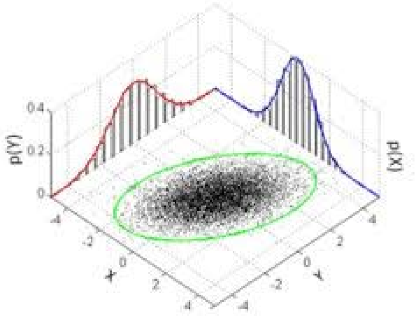


2-d Gaussian

- Bivariate Gaussian distribution
  - Two dimensional random variable: $X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \sim N(\boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \Sigma = \begin{pmatrix} \sigma_1^2 & \sigma(X_1, X_2) \\ \sigma(X_1, X_2) & \sigma_2^2 \end{pmatrix})$$

  - μ_1 *and* μ_2 are means of X_1 *and* X_2
  - σ_1 *and* σ_2 *are* standard deviations of X_1 *and* X_2
  - $\sigma(X_1, X_2)$ *is the covariance between* X_1 *and* X_2, *i.e.*, $\sigma(X_1, X_2) = E(X_1 - \mu_1)(X_2 - \mu_2)$



Apply EM algorithm: 2-d

- An iterative algorithm (at iteration t+1)
  - E(expectation)-step
    - Evaluate the weight w_{ij} when $\boldsymbol{\mu}_j, \Sigma_j, w_j$ are given
    - $w_{ij}^{t+1} = \frac{w_j^t p(\boldsymbol{x}_i | \boldsymbol{\mu}_j^t, \Sigma_j^t)}{\sum_j w_j^t p(\boldsymbol{x}_i | \boldsymbol{\mu}_j^t, \Sigma_j^t)}$
  - M(maximization)-step
    - Find $\boldsymbol{\mu}_j, \Sigma_j, w_j$ that maximize the weighted likelihood, where w_{ij}'s are weights: $\sum_{ij} w_{ij}^{t+1} log w_j p(\boldsymbol{x}_i | \boldsymbol{\mu}_j, \Sigma_j)$
    - It is equivalent to Gaussian distribution parameter estimation when each point has a weight belonging to each distribution
      - $\boldsymbol{\mu}_j^{t+1} = \frac{\sum_i w_{ij}^{t+1} \boldsymbol{x}_i}{\sum_i w_{ij}^{t+1}}$; $(\sigma_{j,1}^2)^{t+1} = \frac{\sum_i w_{ij}^{t+1} \left\| x_{i,1} - \mu_{j,1}^{t+1} \right\|^2}{\sum_i w_{ij}^{t+1}}$; $(\sigma_{j,2}^2)^{t+1} = \frac{\sum_i w_{ij}^{t+1} \left\| x_{i,2} - \mu_{j,2}^{t+1} \right\|^2}{\sum_i w_{ij}^{t+1}}$;
      - $(\sigma(X_1, X_2)_j)^{t+1} = \frac{\sum_i w_{ij}^{t+1} (x_{i,1} - \mu_{j,1}^{t+1})(x_{i,2} - \mu_{j,2}^{t+1})}{\sum_i w_{ij}^{t+1}}$; $w_j^{t+1} \propto \sum_i w_{ij}^{t+1}$

K-Means: A Special Case of Gaussian Mixture Model

- When each Gaussian component with covariance matrix $\sigma^2 I$, and with the same size w_j
  - Soft K-means
  - $w_{ij} \propto p(x_i | \mu_j, \sigma^2) w_j \propto \exp\left\{-\frac{(x_i - \mu_j)^2}{2\sigma^2}\right\} w_j$

Distance!

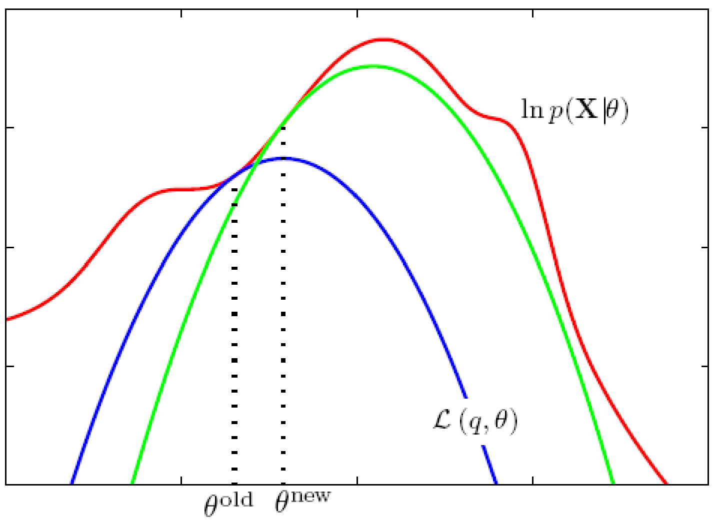
- When $\sigma^2 \rightarrow 0$
  - Soft assignment becomes hard assignment
  - $w_{ij} \rightarrow 1, if\ x_i$ is closest to μ_j (why?)

Mapping Soft Clustering to Hard Clustering

- For evaluation purpose
  - $\mathrm{j}^* = \mathrm{argmax_j w_{ij}}$
  - $w_{ij^*} = 1; w_{ij} = 0\ for\ all\ other\ j \neq j^*$
- Example:
  - $K = 3; the\ output\ of\ GMM\ for\ object\ i\ is$
    - $w_{i1} = 0.7, w_{i2} = 0.2, w_{i3} = 0.1$
    - $\Rightarrow mapping\ result:\ assign\ i\ to\ cluster\ 1$

Why EM Works?*

- E-Step: computing a **tight** lower bound L of the original objective function l at θ_{old}
- M-Step: find θ_{new} to maximize the lower bound
- $l(\theta_{new}) \geq L(\theta_{new}) \geq L(\theta_{old}) = l(\theta_{old})$



How to Find Tight Lower Bound?*

- $$\begin{aligned} \ell(\theta) &= \log \sum_h p(d,h;\theta) \\ &= \log \sum_h \frac{q(h)}{q(h)} p(d,h;\theta) \\ &= \log \sum_h q(h) \frac{p(d,h;\theta)}{q(h)} \end{aligned}$$

$q(h)$: the key to tight lower bound we want to get

- Jensen's inequality
  - $$\log \sum_h q(h) \frac{p(d,h;\theta)}{q(h)} \geq \sum_h q(h) \log \frac{p(d,h;\theta)}{q(h)}$$

the tight lower bound

- When "=" holds to get a tight lower bound?
  - $q(h) = p(h|d,\theta)$ (why?)

In GMM Case*

$$L(D;\theta) = \sum_i \log \sum_j w_j p(x_i|\mu_j, \sigma_j^2)$$

$$\geq \sum_i \sum_j w_{ij} \left(\underbrace{\log w_j p\left(x_i \middle| \mu_j, \sigma_j^2\right)}_{\log L(x_i, z_i = j|\theta)} - \underbrace{log w_{ij}}_{\text{Does not involve } \theta \text{, can be dropped}}\right)$$

$\log L(x_i, z_i = j|\theta)$

Does not involve θ,
can be dropped

Advantages and Disadvantages of GMM

- Strength
  - Mixture models are more general than partitioning: different densities and sizes of clusters
  - Clusters can be characterized by a small number of parameters
  - The results may satisfy the statistical assumptions of the generative models
- Weakness
  - Converge to local optimal (overcome: run multi-times w. random initialization)
  - Computationally expensive if the number of distributions is large
  - Hard to estimate the number of clusters
  - Can only deal with spherical clusters

Vector Data: Mixture Model

- Revisit K-means
- Mixture Model and EM algorithm
- Summary

Summary

- Revisit k-means
  - Limitations
- Mixture models
  - Gaussian mixture model; multinomial mixture model; EM algorithm; Connection to k-means