Fundamentals of Machine Learning

UNSUPERVISED LEARNING - CLUSTERING

Amit K Roy-Chowdhury

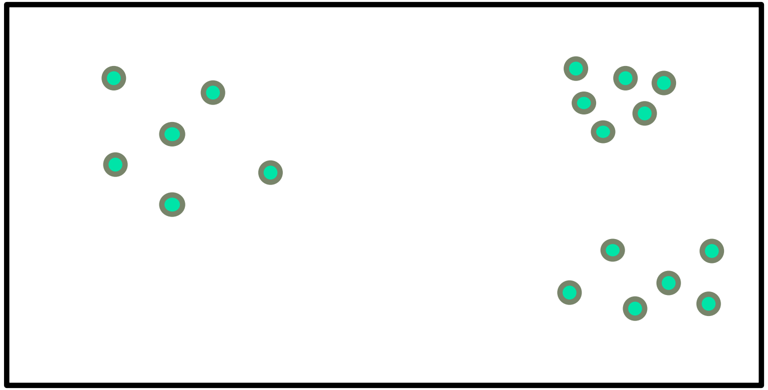
Acknowledgments: Adapted from slides at https://probml.github.io/pml-book/teaching1.html by Prof. Saw Shier Nee
Modified by Amit Roy-Chowdhury, UCR.

Outline

- Dimensionality Reduction methods (PCA) – already covered
- Clustering algorithms – K-means
- Gaussian Mixture Models – EM Algorithm

Clustering

- Unsupervised Learning
- No Ground Truth
- Investigate data structure by grouping them into distinct groups.
- Advantage: useful when don't know what to look for
- Disadvantage: Subjective



Applications

- Find set of representative samples that provide coverage of the data
  - Identify fake news based on the content
  - Group emails

- Exploratory Insight
  - Customer shopping patterns
  - Groups of genes/proteins with similar function

Clustering

Basic idea: group similar instances together

Example: 2D point patterns

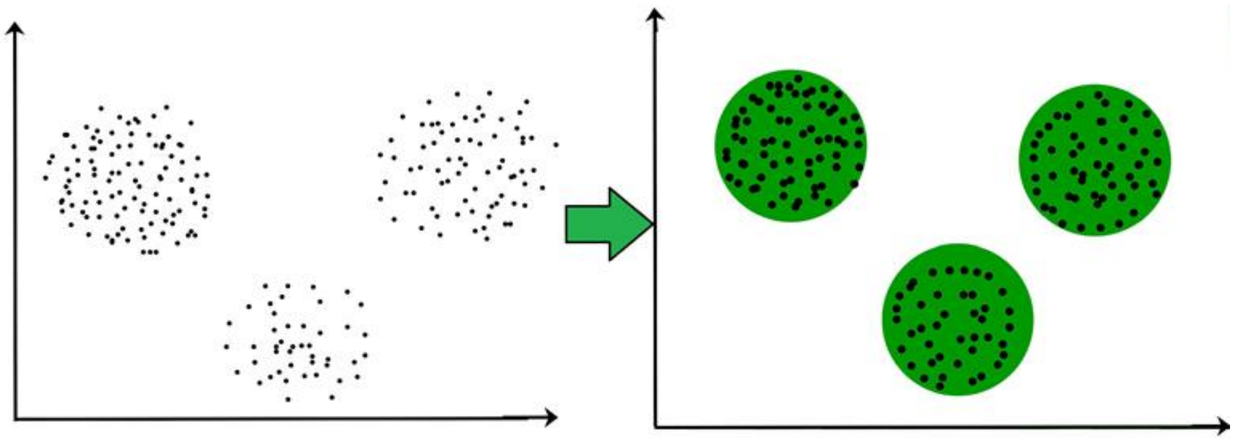
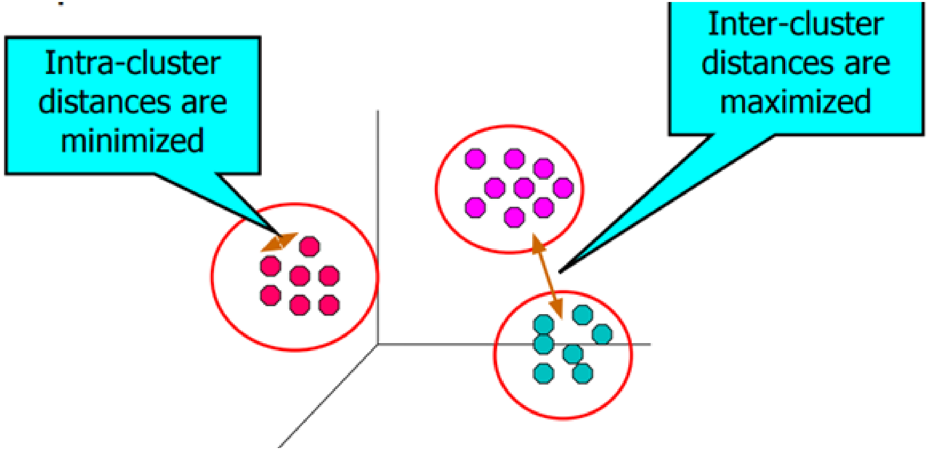


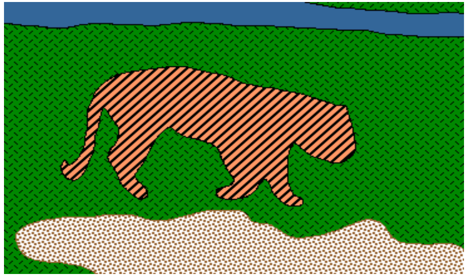
Image taken from https://www.geeksforgeeks.org/clustering-in-machine-learning/

Clustering

- Basic idea: group similar instances together
- What does "similar" mean?
    - Based on Euclidean distance (squared),

      or
    - Measure of similarity (or distance) between "points" to be clustered



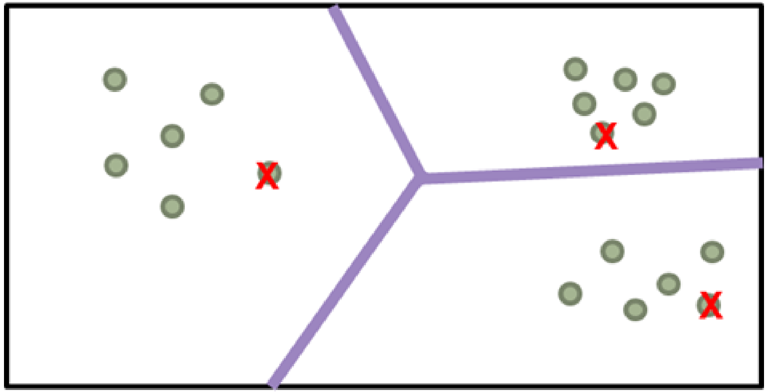


Gestalt Laws seek to formalize this proximity, similarity, continuation, closure, common fate

Clustering

- Partitional - data in non-overlapping subsets. One data object is in one subset
- Hierarchy - data are in nested clusters, organized in a hierarchical tree

<u>Partitional</u>



<u>Hierarchy</u>

Bourns College of Engineering

Kmeans Algorithm

Also known as Lloyd's algorithm

Input: $x_1, x_2, \ldots, x_n$, given $x \in R^d$

Output: 'Centers', $\mu_1, \mu_2, \ldots, \mu_k$, given $\mu \in R^d$

Assign each data point to its closest centers: $z_n^* = \arg\min_k ||\boldsymbol{x}_n - \boldsymbol{\mu}_k||_2^2$

Compute the cluster centers: $$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n:z_n=k} \boldsymbol{x}_n$$

Iterate between them till convergence – closely related to Expectation Maximization

Summary - Kmeans

A particularly simple method for clustering is K-means, which is identical to the generalized Lloyd algorithm we know from vector quantization, just applied to clustered data.

The idea is to represent each cluster k by a center point c_k and assign each data point x_n to one of the clusters k, which can be written in terms of index sets C_k.

The center points and the assignment are then chosen such that the mean squared distance between data points and center points is minimized

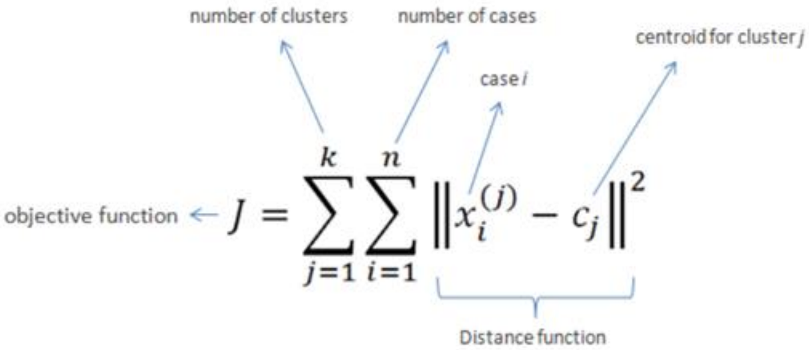
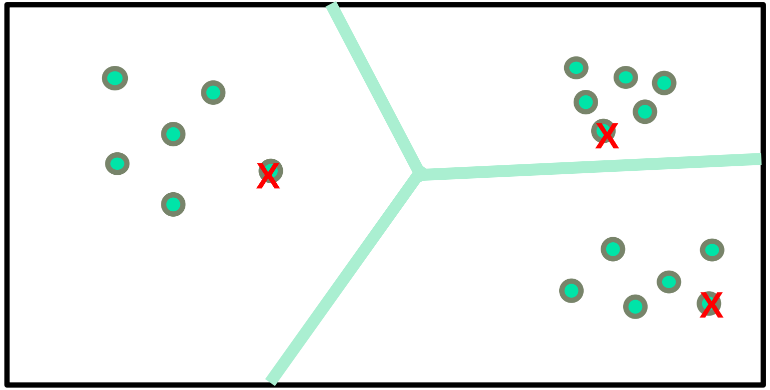


Image from https://www.saedsayad.com/clustering_kmeans.htm

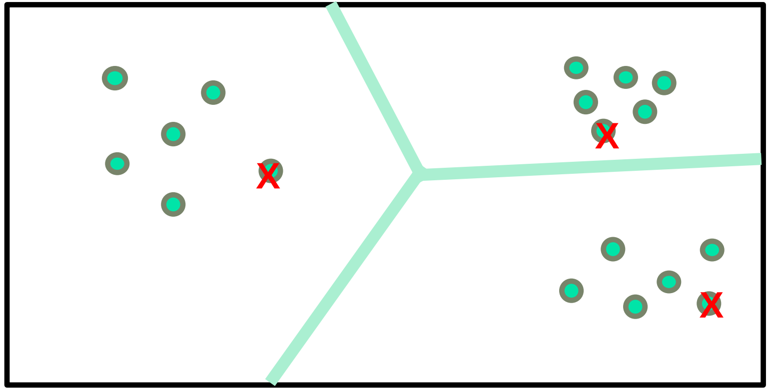
K-means - Intuition

Say we want three clusters, first initialize random centre



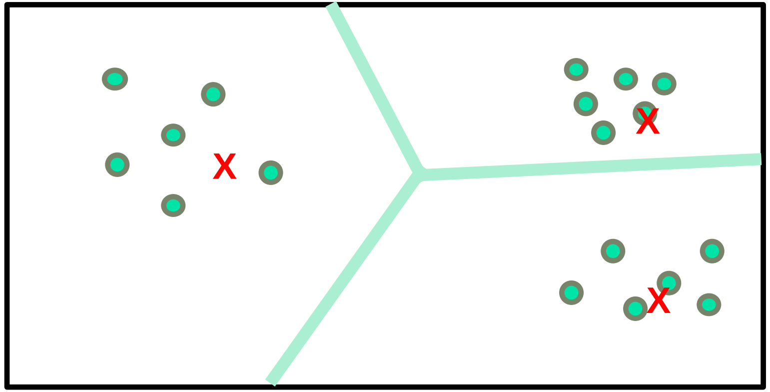
Generate optimal partition

K-means - Intuition



Update the centre by computing the mean of coordinates in their respective region.

K-means - Intuition

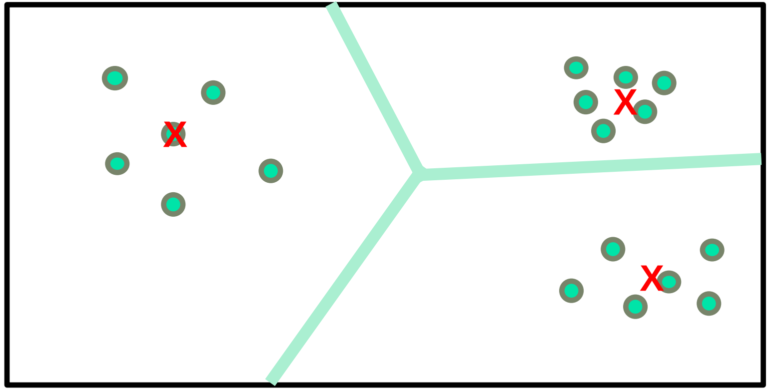


Updated centre

Compute Euclidean distance between points to the closest centres.

Identify the three regions

K-means - Intuition



Centre no longer changes (converge). Now, we have identify the three clusters.

K-means Algorithm

K-means clustering algorithm

1. Randomly initialize the cluster centers, $c_1, \ldots, c_K$
2. Given cluster centers, determine points in each cluster
   - For each point p, find the closest c_i. Put p into cluster i
3. Given points in each cluster, solve for c_i
   - Set c_i to be the mean of points in cluster i
4. If c_i have changed, repeat Step 2

Properties

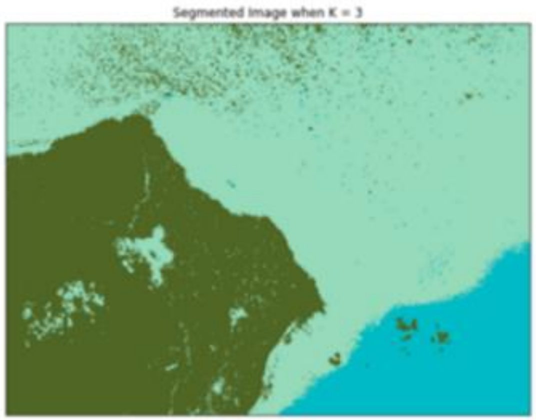
Will always converge to *some* solution

Can be a "local minimum"

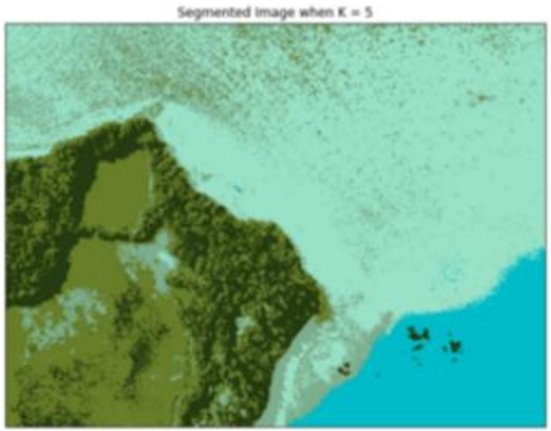
- does not always find the global minimum of objective function:

$$\sum_{\text{clusters } i} \; \sum_{\text{points p in cluster } i} ||p - c_i||^2$$

Kmeans for Segmentation



K=3

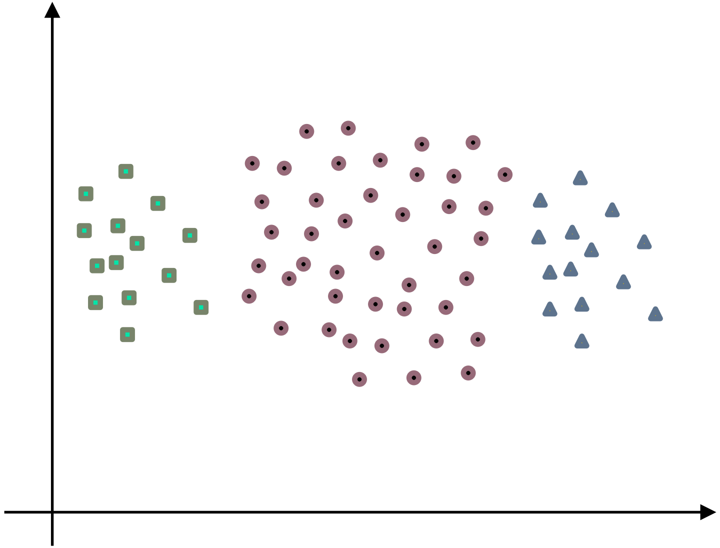


K=5

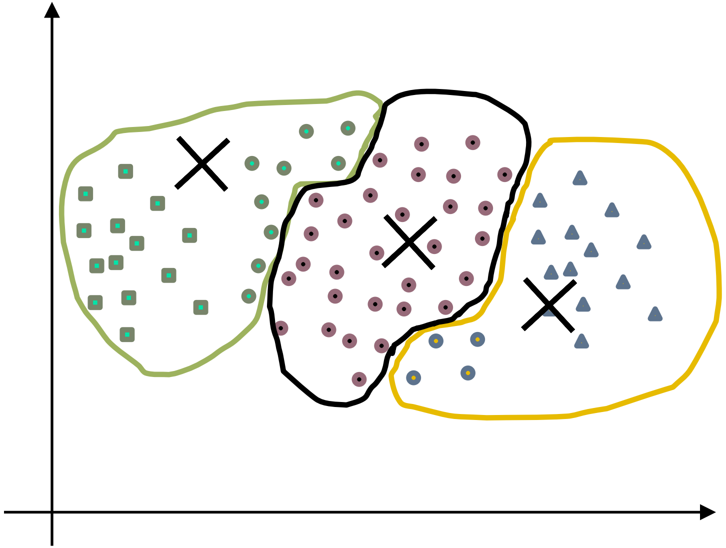
Limitation

- K-means has problems when clusters are of different
  - Sizes
  - Densities
  - Non-globular shapes
- K-means has problems when the data contains outliers.

Limitation – Different Size

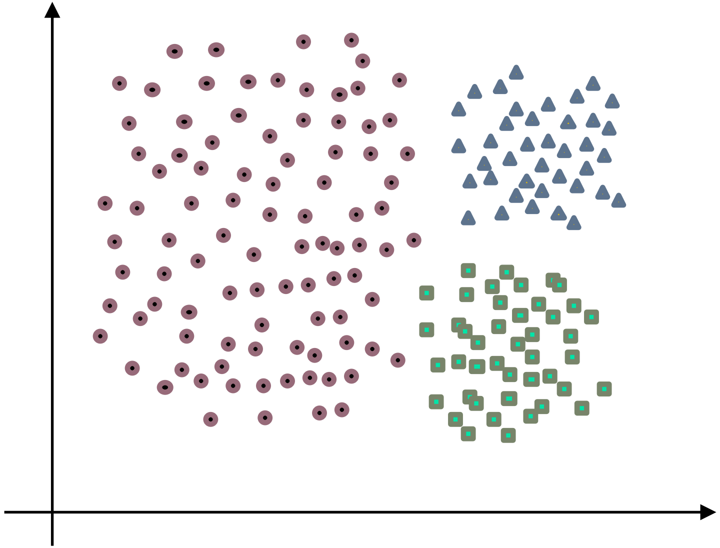


Data point

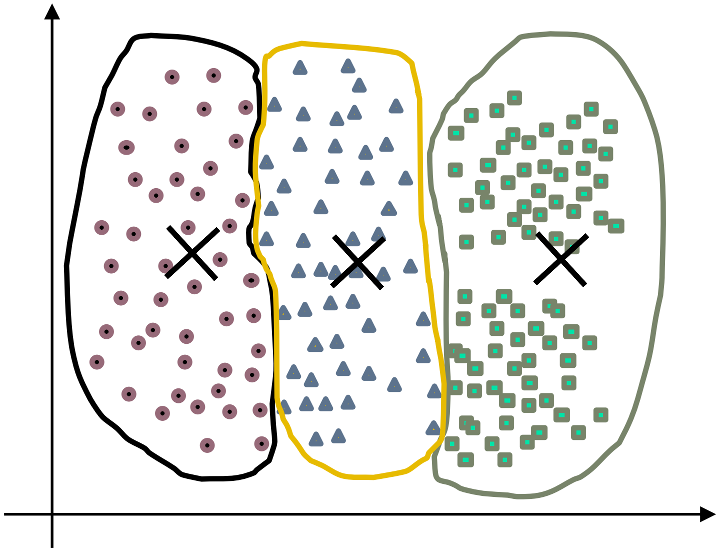


K = 3

Limitation – Different Density

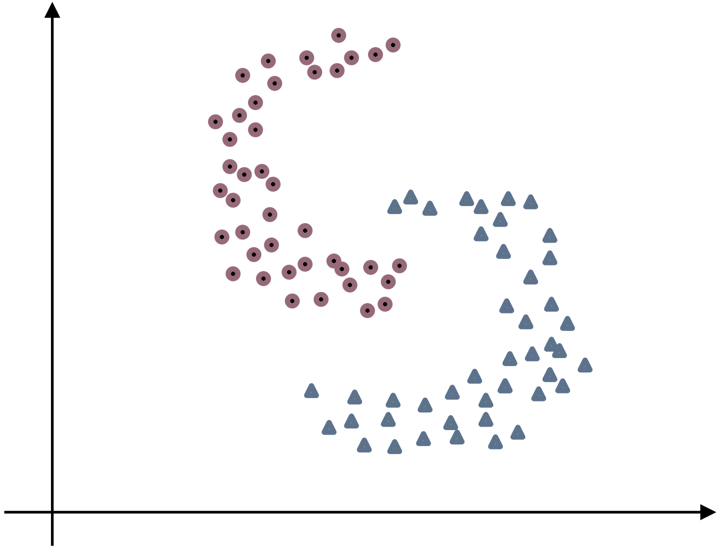


Data point

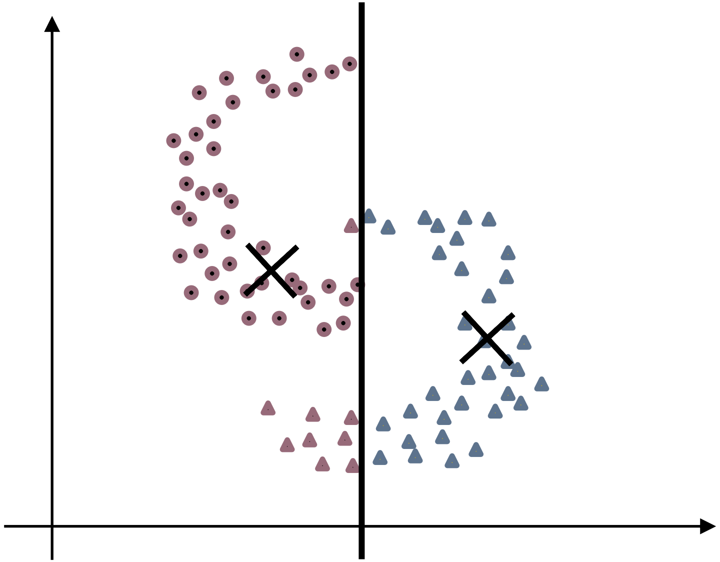


K = 3

Limitation – non globular shape



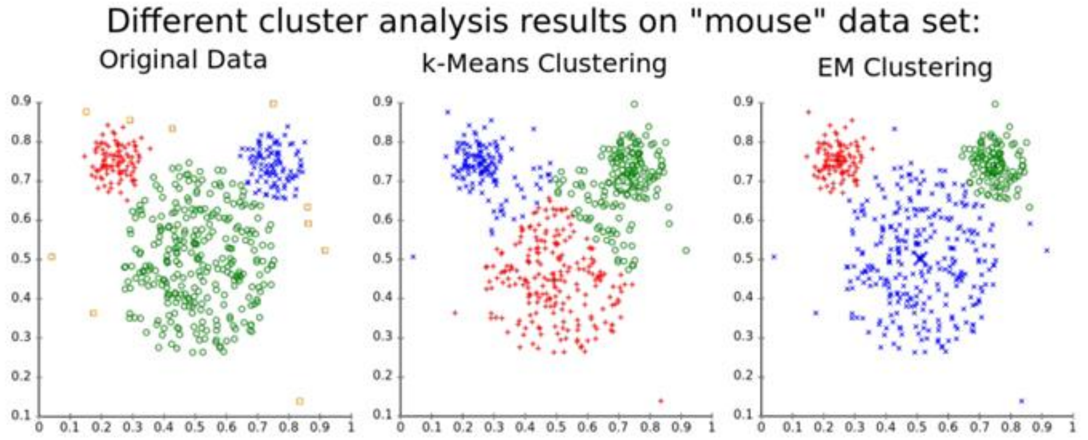
Data point



K = 2

To overcome k-means limitation

- Generate more clusters. Then combine similar cluster
- EM clustering using Gaussian Mixture Model
  - Intuition: Assume that the data generating process is a mixture of Multivariate Normals

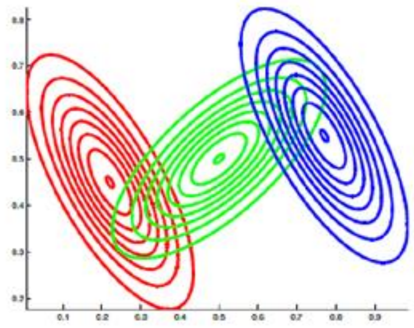


[source: Wikipedia, Public Domain image]

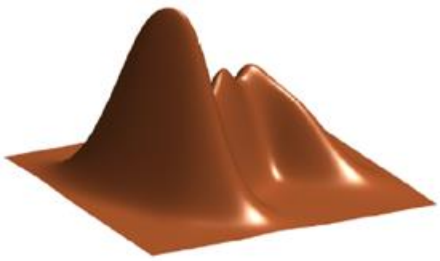
Gaussian Mixture Model/Mixture of Gaussian

$$p(y|\theta) = \sum_{k=1}^{K} \pi_k p_k(y)$$ where p_k is the k'th mixture component, and π_k are the mixture weights which satisfy $0 \le \pi_k \le 1$ and $\sum_{k=1}^{K} \pi_k = 1$.

$$p(y|\theta) = \sum_{k=1}^{K} \pi_k \mathcal{N}(y|\mu_k, \Sigma_k)$$



A mixture of 3 Gaussians in 2d.



A surface plot of the overall density.

Expectation Maximization

$\mathcal{D} = \{y_n : n = 1 : N\}$

Need to find MLE $\hat{\theta} = \operatorname{argmax} \log p(\mathcal{D}|\theta)$

Approach:

1. E step: given current parameters, compute ownership of each point (hidden variable)
2. M step: given ownership probabilities, update parameters to maximize likelihood function
3. repeat until convergence

Expectation Maximization - Details

E Step: responsibility of cluster k for generating data point n

$$r_{nk}^{(t)} = p^*(z_n = k | y_n, \boldsymbol{\theta}^{(t)}) = \frac{\pi_k^{(t)} p(y_n | \boldsymbol{\theta}_k^{(t)})}{\sum_{k'} \pi_{k'}^{(t)} p(y_n | \boldsymbol{\theta}_{k'}^{(t)})}$$

M Step: maximizes log likelihood

$$\boldsymbol{\mu}_k^{(t+1)} = \frac{\sum_n r_{nk}^{(t)} \boldsymbol{y}_n}{r_k^{(t)}}$$

$$\boldsymbol{\Sigma}_k^{(t+1)} = \frac{\sum_n r_{nk}^{(t)} (\boldsymbol{y}_n - \boldsymbol{\mu}_k^{(t+1)})(\boldsymbol{y}_n - \boldsymbol{\mu}_k^{(t+1)})^{\mathsf{T}}}{r_k^{(t)}}$$

$$\pi_k^{(t+1)} = \frac{1}{N} \sum_n r_{nk}^{(t)} = \frac{r_k^{(t)}}{N}$$

$r_k^{(t)} \triangleq \sum_n r_{nk}^{(t)}$ is the weighted number of points assigned to cluster k

Relation to K-Means

we fix $\Sigma_k = \mathbf{I}$ and $\pi_k = 1/K$ for all the clusters (so we just have to estimate the means $\boldsymbol{\mu}_k$)

we approximate the E step, by replacing the soft responsibilities with hard cluster assignments

$$z_n^* = \operatorname{argmax}_k r_{nk}$$

EM clustering