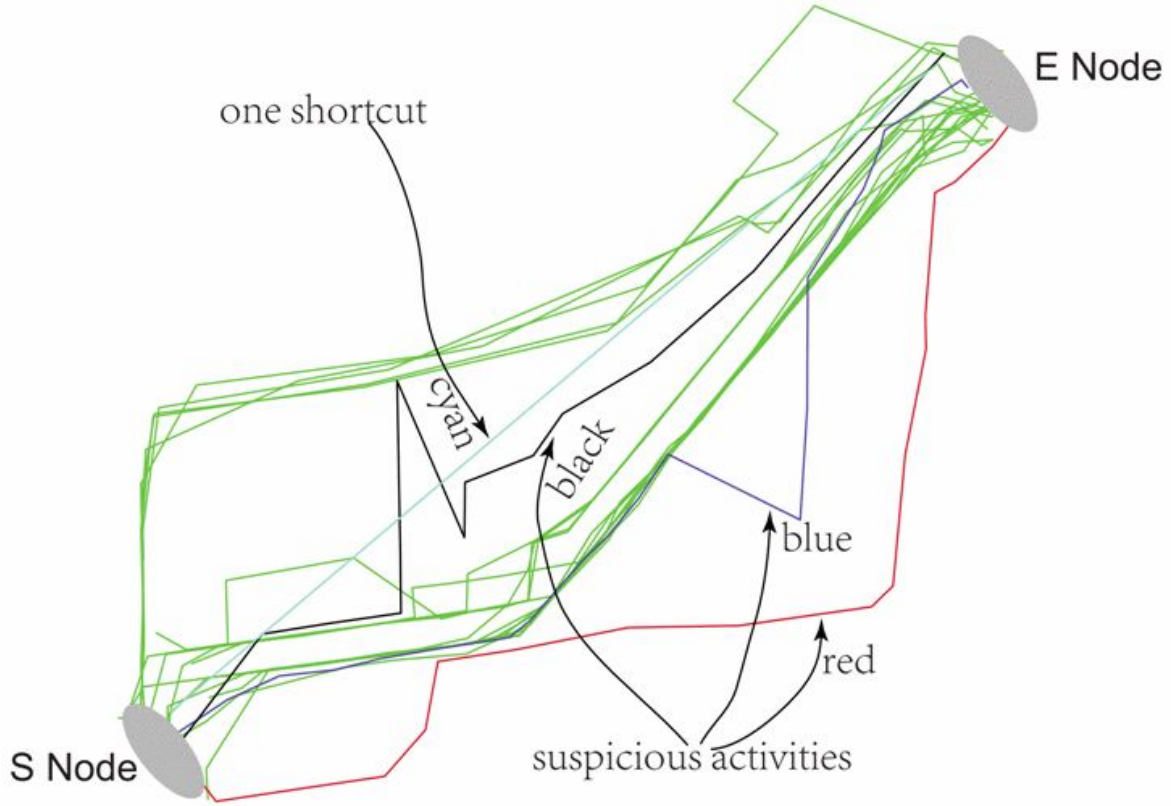
Clustering

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

What is Clustering

- Organizing data into clusters such that there is
  - high intra-cluster similarity
  - low inter-cluster similarity
- Informally, finding natural groupings among objects

An interesting Use Case: Fraudulent Driving



A Taxi Driving Fraud Detection System

Unsupervised Learning

- Clustering methods are unsupervised learning techniques
  - We do not have a teacher that provides examples with their labels
    - Requires data, but no labels
- Detect patterns e.g. in
  - Group emails or search results
  - Customer shopping patterns
  - Regions of images
- Useful
  - when don’t know what you’re looking for
  - Unavailability of the annotations

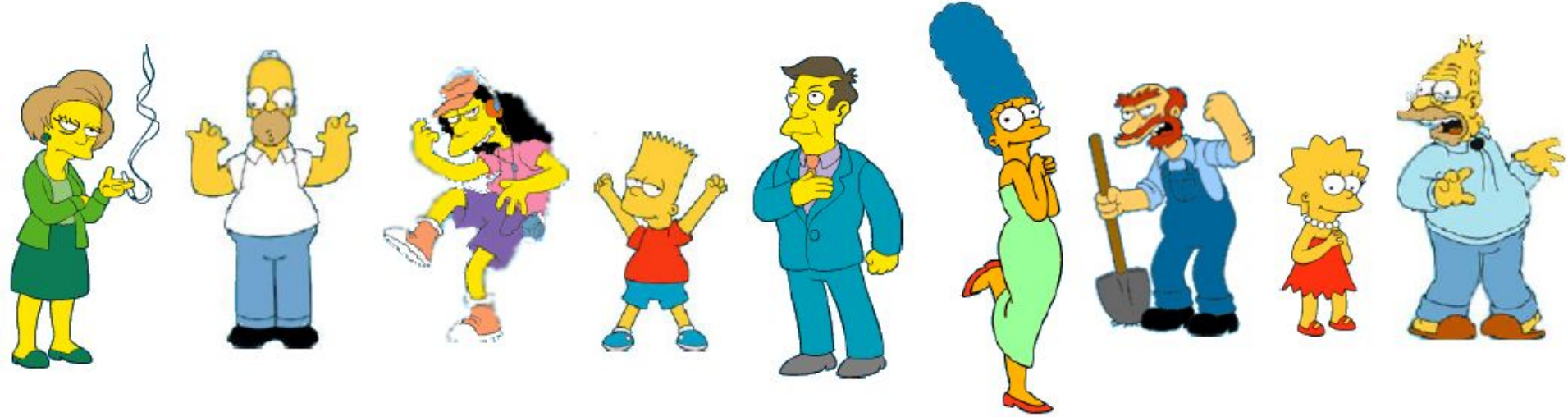
Why Clustering

- Organizing data into clusters provides information about the internal structure of the data
  - Ex. detecting fraudulent driving

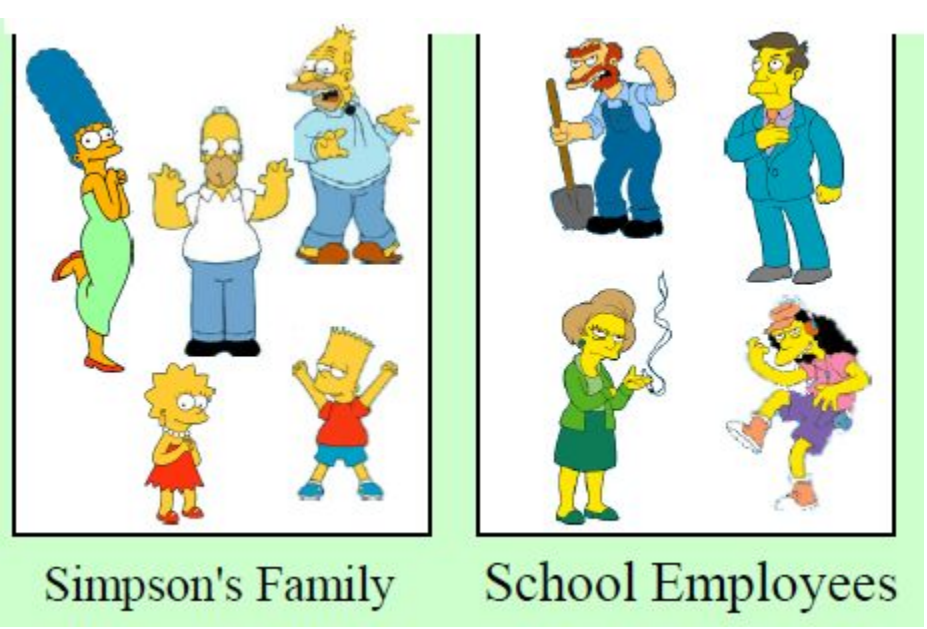
- Sometimes the partitioning is the goal
  - Image segmentation
  - Places which are contributing most to the pollution

- Knowledge discovery in data
  - Underlying rules, recurring patterns, topics, etc

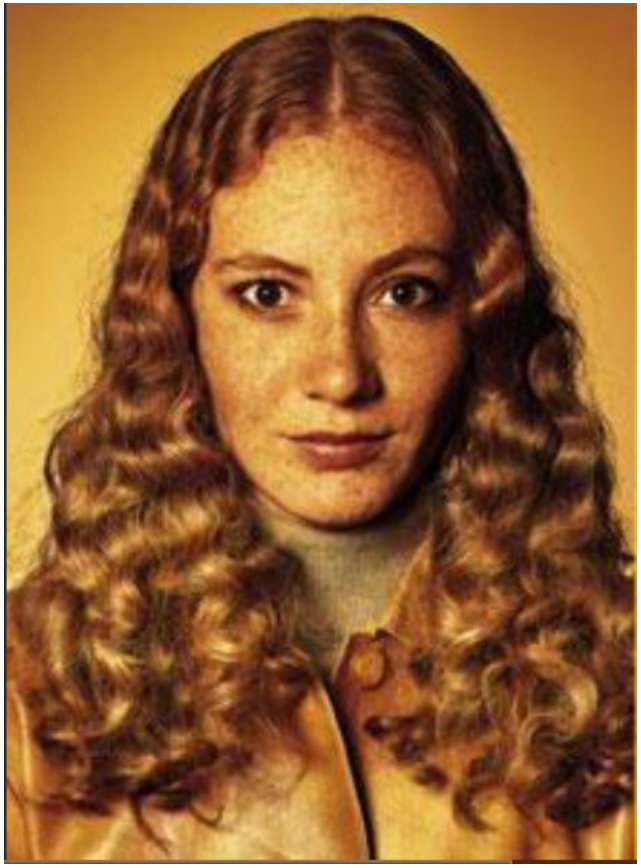
Natural grouping : Clustering is subjective



Natural grouping : Clustering is subjective



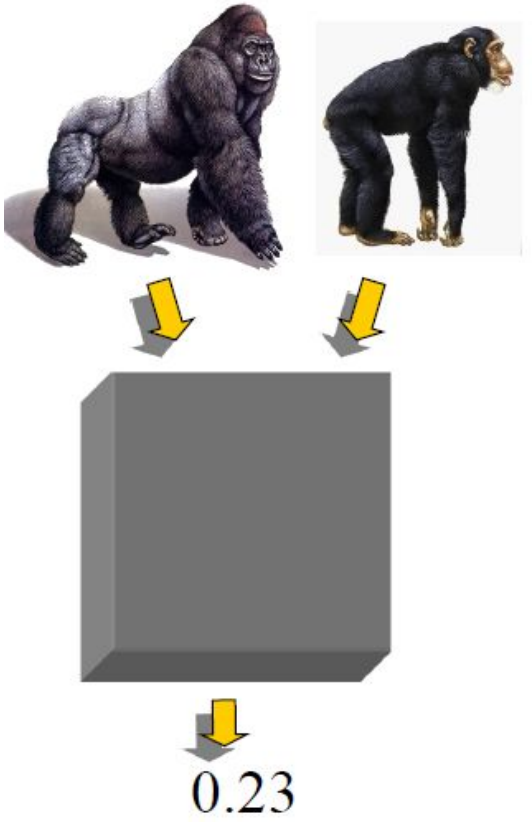
What is Similarity?



Distance Measures

- The distance (dissimilarity) between O_1 and O_2 is a real number denoted by $D(O_1,O_2)$.

- Euclidean distance
  - $d(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$
- Correlation coefficient
  - $s(x, y) = \sum_i (x_i - \mu_x)(y_i - \mu_y)/\sigma_x\sigma_y$

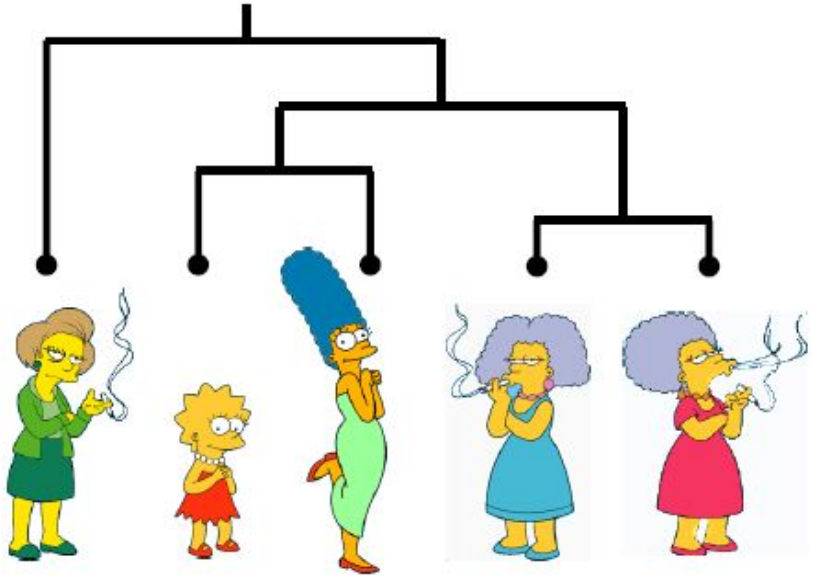


Desirable Properties

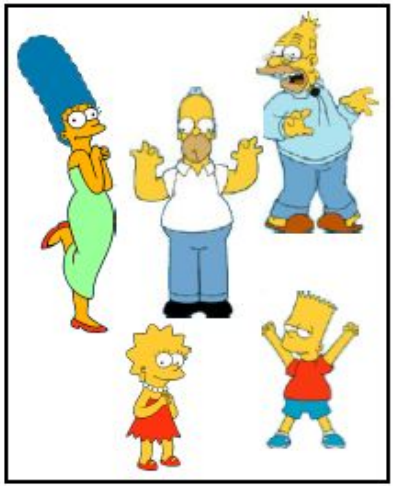
- Symmetric
  - D(A, B) = D (B,A)
  - Otherwise, we can say that A looks like B but B does not look like A
- Positivity and Self-similarity
  - D(A, B) >=0 and D (A, B) = 0, iff A = B
  - Otherwise, there will be different objects that we cannot tell apart
- Triangle Inequality
  - D(A, B) + D(B, C) >= D(A, C)
  - Otherwise, we can say that A is like B, B is like C but A is not like C at all

Clustering Types

Hierarchical



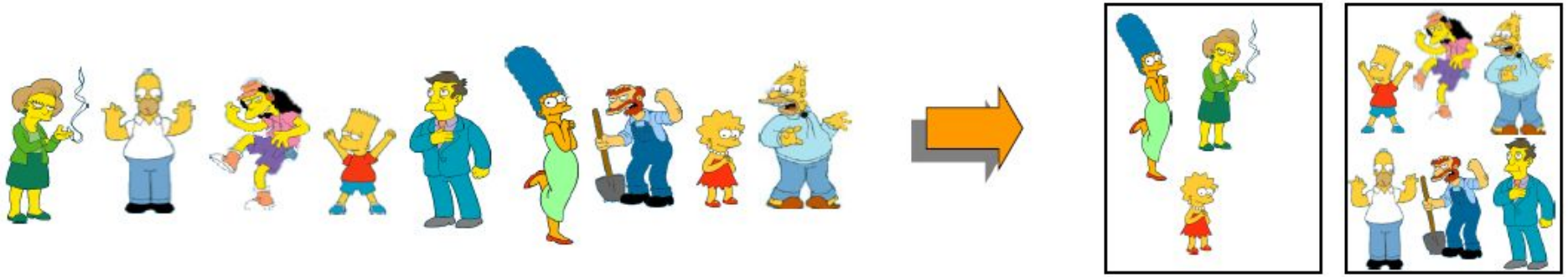
Partitional



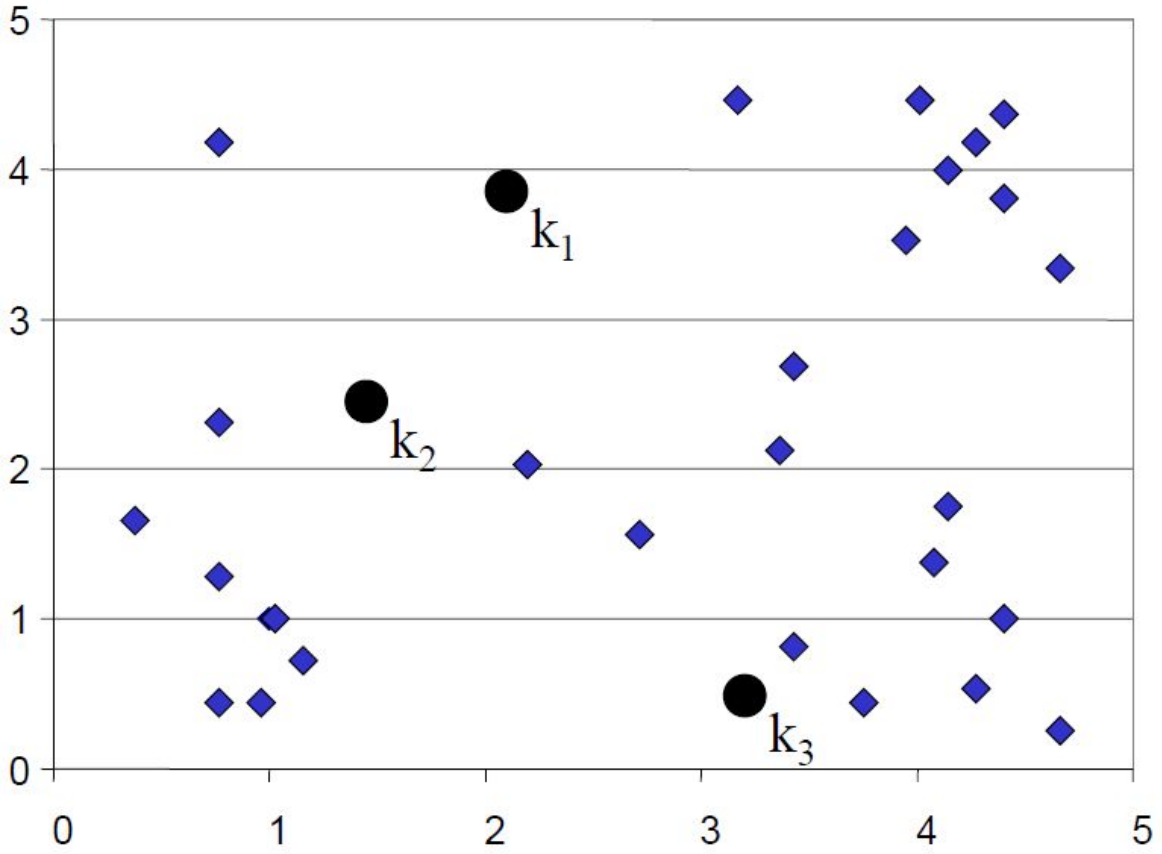
Partitional Clustering: K-Means Clustering

- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid
- Number of clusters, K, must be specified
- The objective is to minimize the sum of distances of the points to their respective centroid

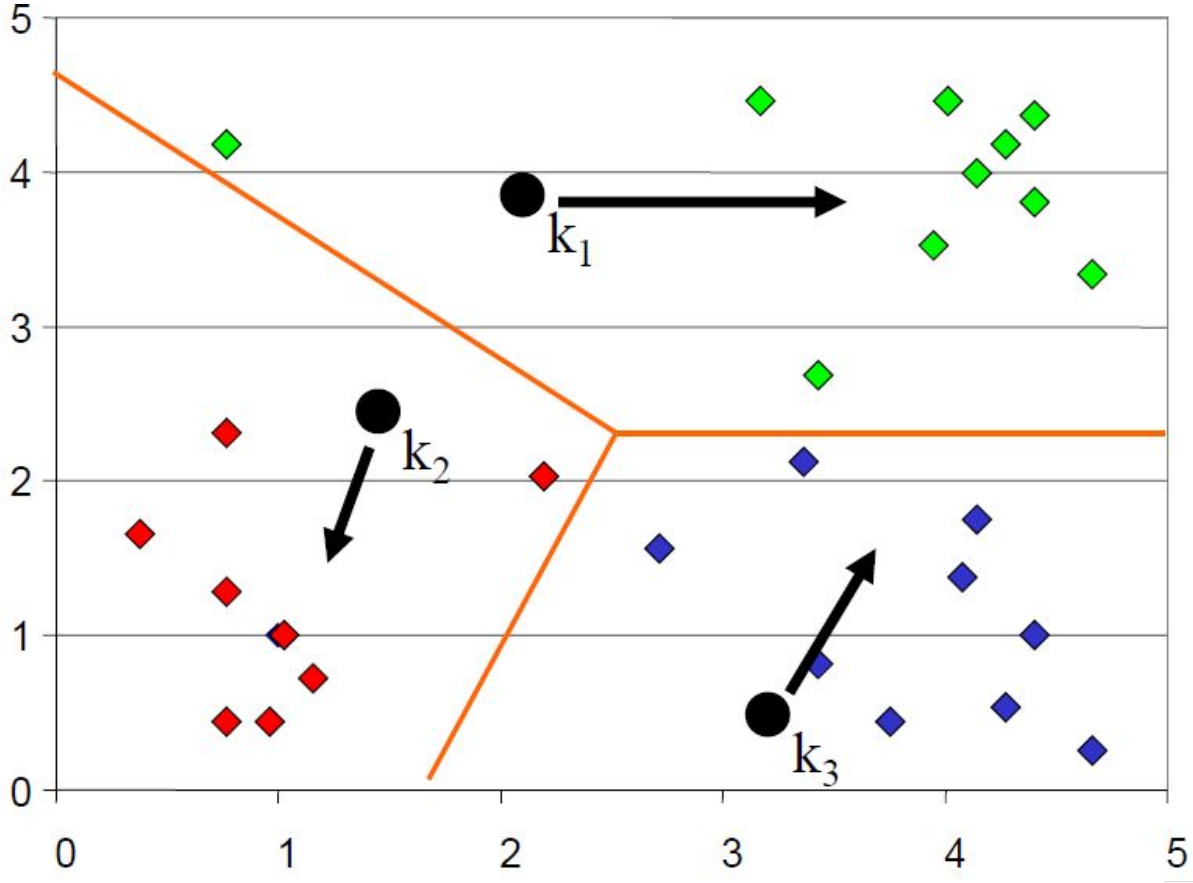
K-Means Clustering



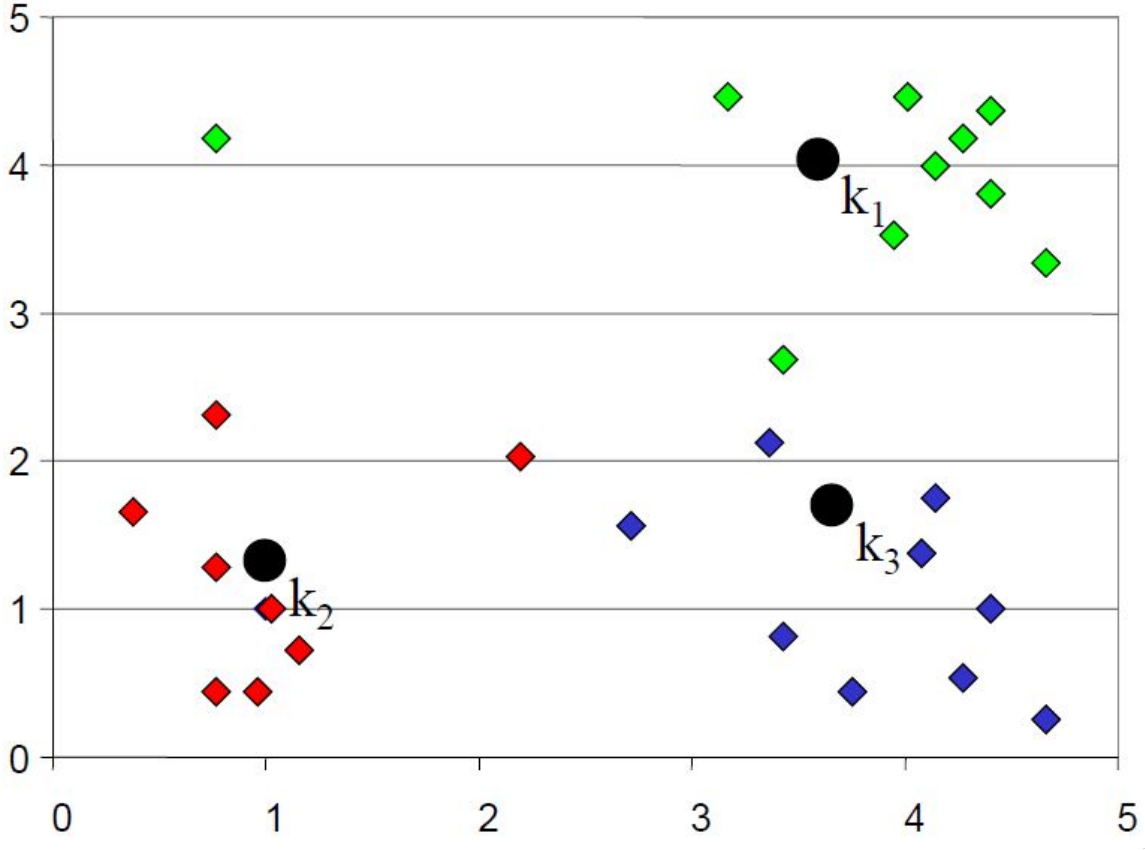
K-Means Clustering: Initialization



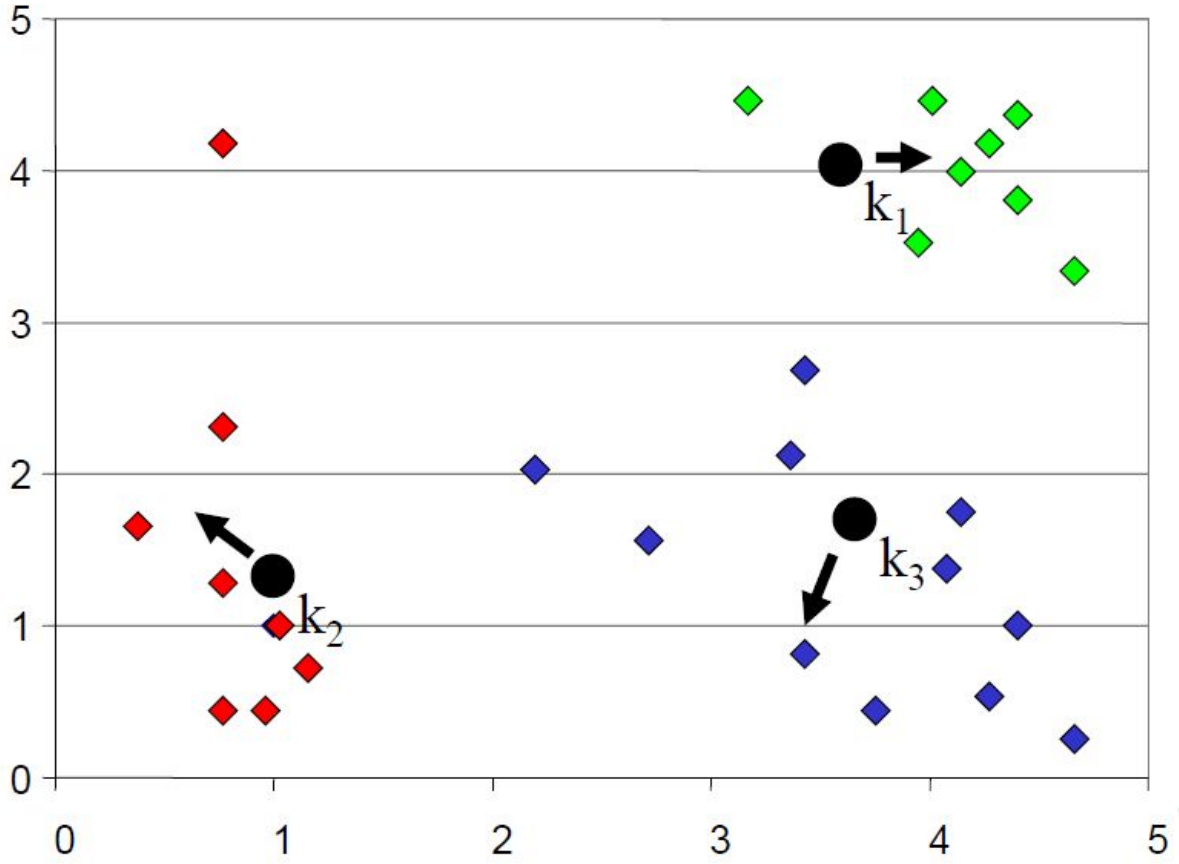
K-Means Clustering: Iteration-I



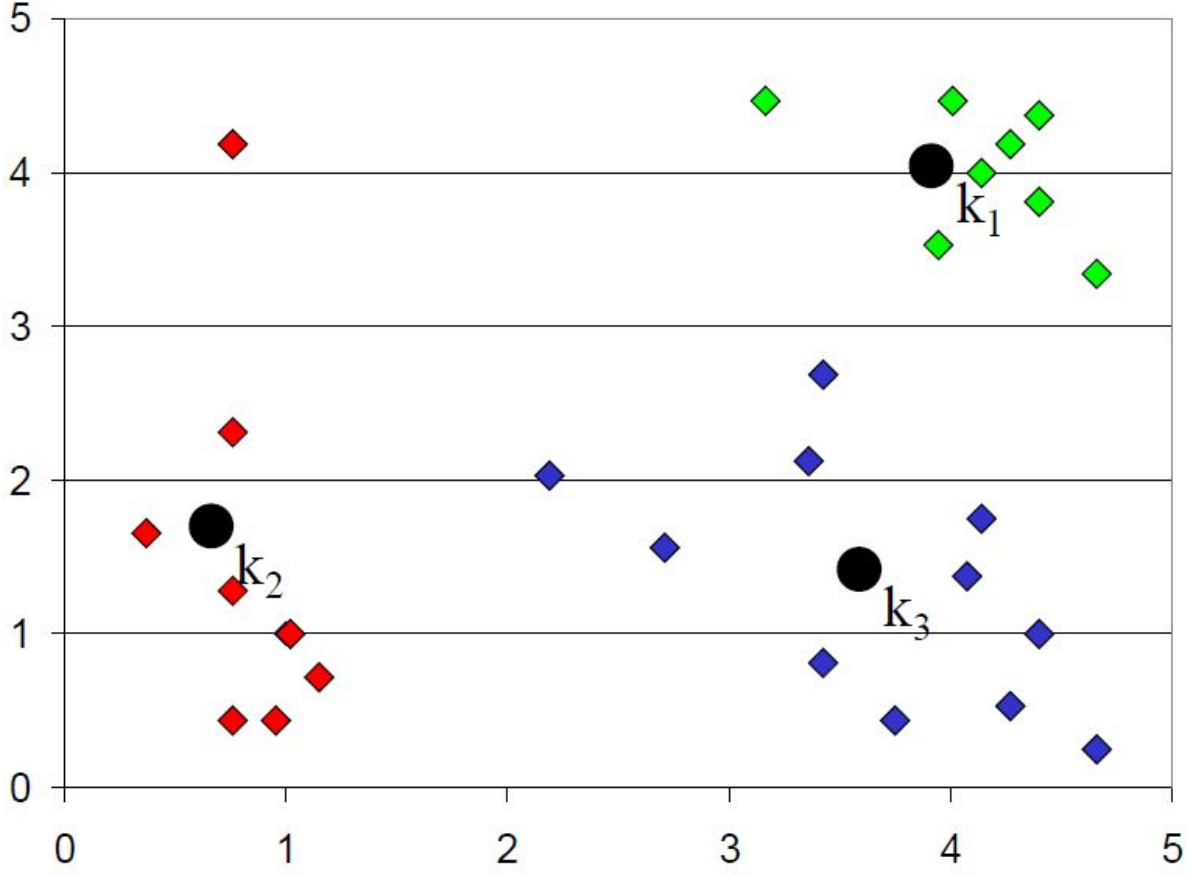
K-Means Clustering: Iteration-I



K-Means Clustering: Iteration-II



K-Means Clustering: Iteration-III



Algorithm: K-Means Clustering

0.1 Decide on a value for K, the number of clusters.
0.2 Initialize the K cluster centers (randomly, if necessary).
1. *Assignment:* Decide the class memberships of the *n* objects by assigning them to the nearest cluster center.

$$min \sum_{i=1}^{n} |x_i - \mu_{x_i}|^2$$

2. *Re-estimate* the K cluster centers, by assuming the memberships found above are correct.

$$\mu_i = \frac{1}{|C_i|} \sum_{x \varepsilon C_i} x$$

Repeat 3 and 4 until none of the *n* objects changed membership in the last iteration.

Convergence: K-Means Clustering

- Loss Function of the K-Means

$$L(C, \mu) = \min_{\mu} \min_{C} \sum_{i=1}^{k} \sum_{x \varepsilon C_i} |x - \mu_i|^2$$

- Fix μ optimize C
  - Assign data points to closest cluster center

$$L(C^{t+1}, \mu^t) < L(C^t, \mu^t)$$

- Fix C optimize μ
  - Change the cluster center to the average of its assigned points

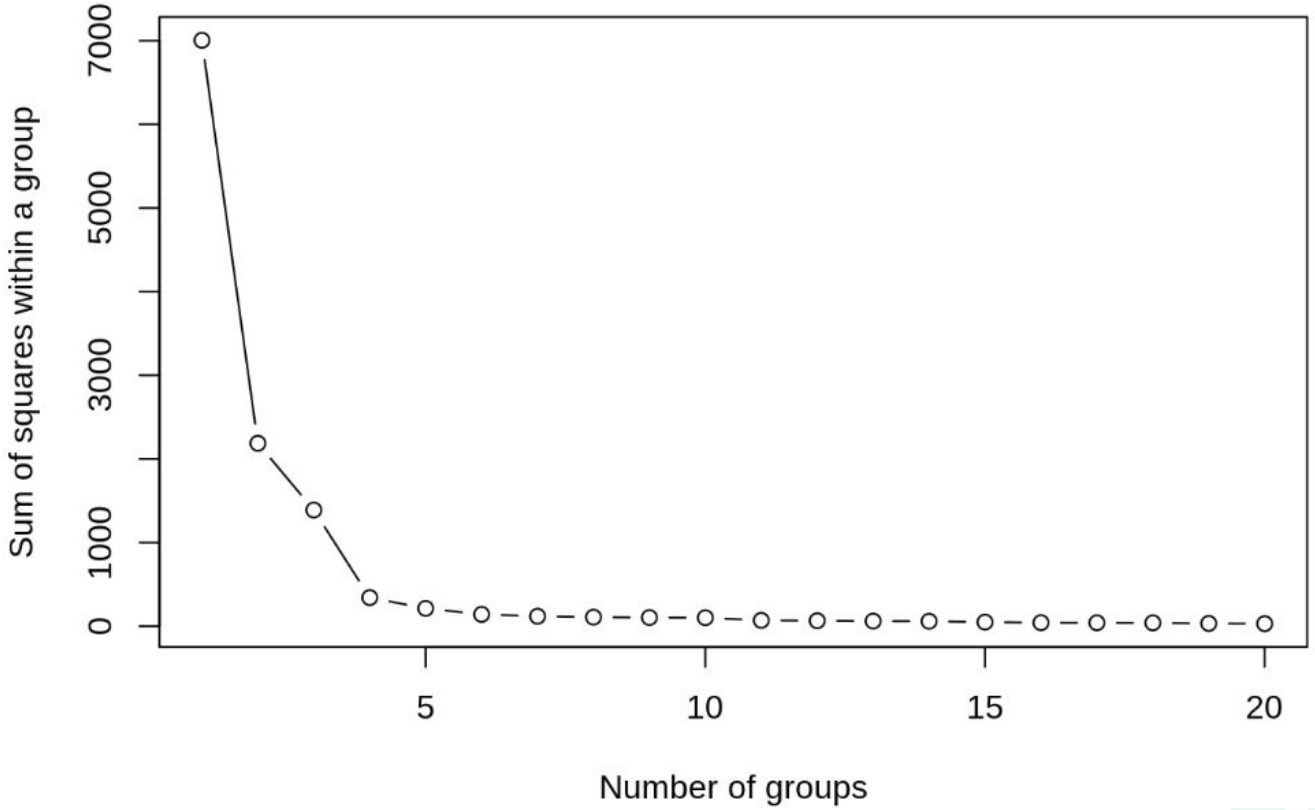
$$L(C^{t+1}, \mu^{t+1}) <= L(C^{t+1}, \mu^t)$$

- Loss function is guaranteed to decrease monotonically in each iteration in each steps until convergence.

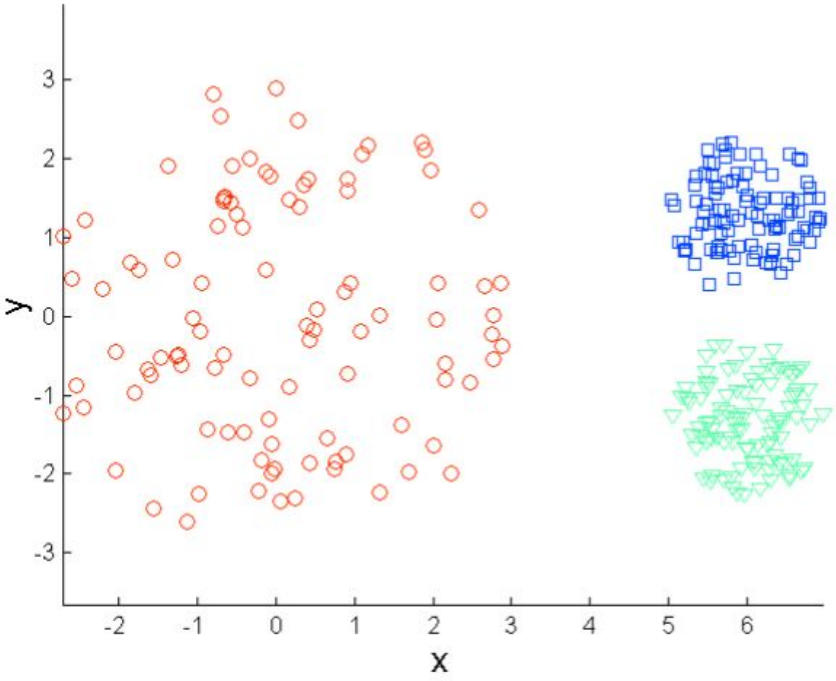
Properties: K-Means Clustering

- Guaranteed to converge in a finite number of iterations
- Running time per iterations:
  - Assign data points to closest cluster center
    - O(KN)
  - Change the cluster center to the average of its assigned points
    - O(N)

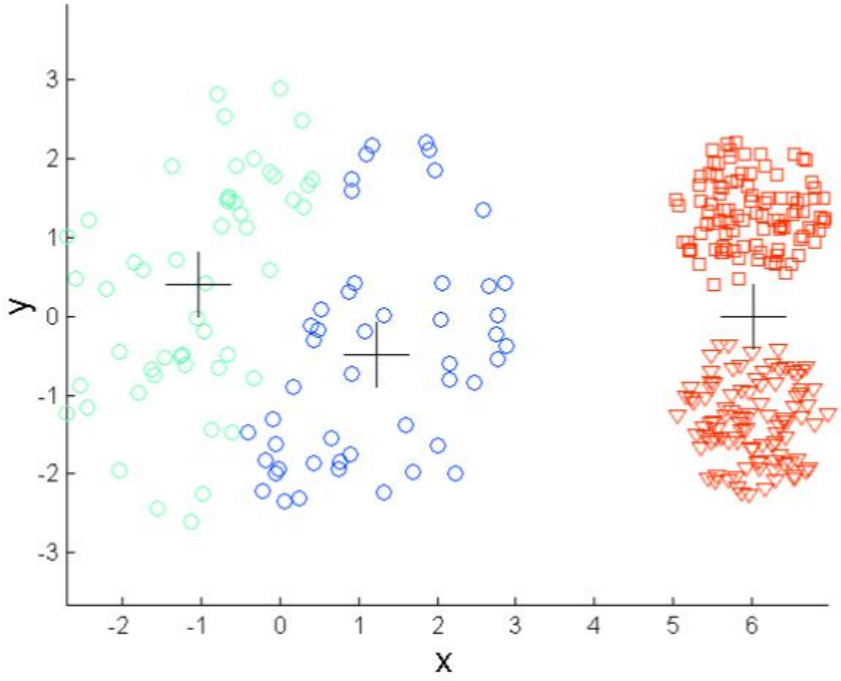
Value of K? $L(C,\mu) = \min_{\mu}\min_{C} \sum_{i=1}^{k} \sum_{x \varepsilon C_i} |x - \mu_i|^2$



Limitations: Different Sizes

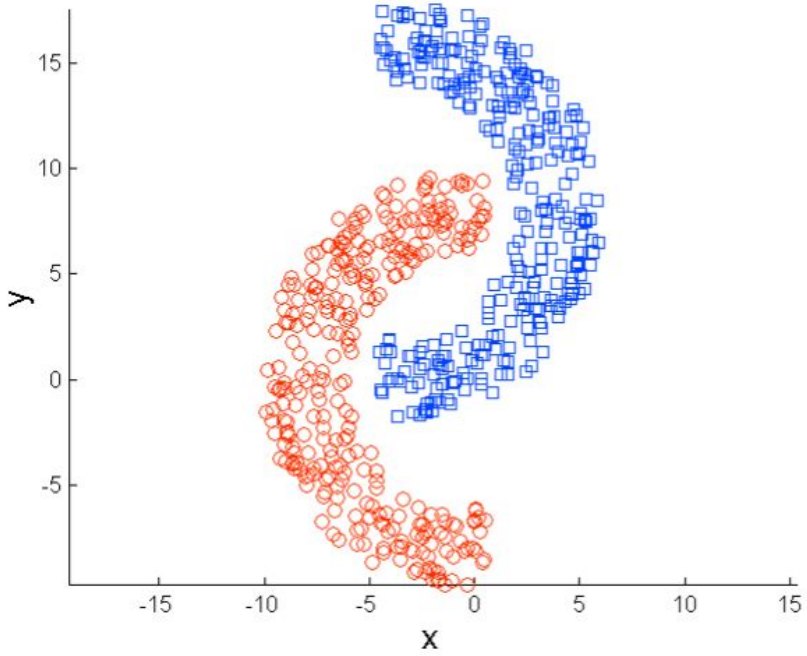


Original Points

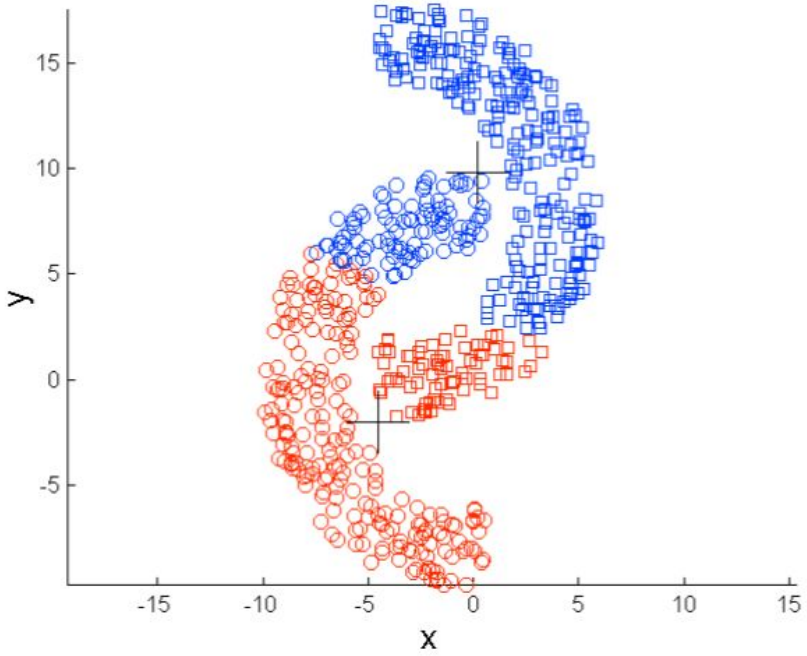


K-means (3 Clusters)

Limitations: Non-convex



Original Points



K-means (2 Clusters)

Summary: K-Means Clustering

- Strength
  - Simple, easy to implement and debug
  - Relatively efficient: O(tkn), where n is # objects, k is # clusters, and t is # iterations. Normally, k, t << n.
- Weakness
  - Applicable only when mean is defined, what about categorical data?
  - Often terminates at a local optimum. Initialization is important.
  - Need to specify K, the number of clusters, in advance
  - Unable to handle noisy data and outliers
  - Not suitable to discover clusters with non-convex shapes

Breakout Room Activity

You are given a 1-d dataset as follows, X = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

a. With number of cluster, k = 2, and initial cluster centroids as {1, 2}, show three iterations of k-means algorithm. Use Euclidean distance function.
b. Repeat above question with initial cluster centroids as {2, 9}. 2 point for each correct iteration.
c. Explain your observations about how the choice of initial seed set affects the quality of results.

References

1. http://www.iro.umontreal.ca/~lisa/pointeurs/kmeans-nips7.pdf
2. https://cs.wmich.edu/alfuqaha/summer14/cs6530/lectures/ClusteringAnalysis.pdf
3. https://ieeexplore.ieee.org/abstract/document/6137222
4. https://home.deib.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html
5. https://www.youtube.com/watch?v=4b5d3muPQmA