DATA ANALYTICS

Unit 4: Brief Review of Unsupervised Learning Algorithms (k-means, Hierarchical agglomerative clustering)

Jyothi R.

Department of Computer Science and Engineering

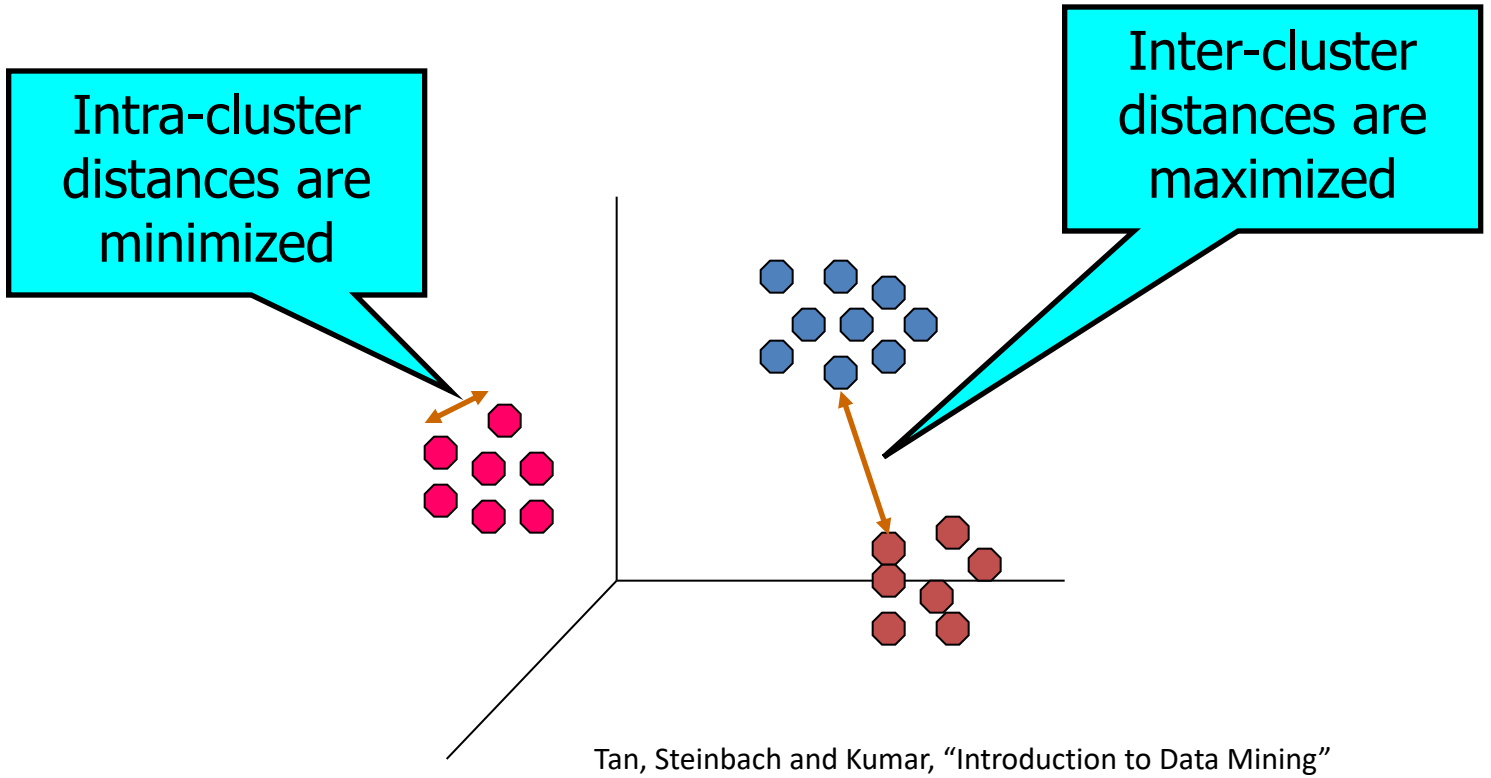
DATA ANALYTICS

Introduction

- Mastering unsupervised learning opens up a broad range of avenues for a data scientist.
- Clustering has wide application across domains and industries
- Examples: **Uber's route optimization, Amazon's recommendation system, Netflix's customer segmentation**, and so on…
- Why is it important?
  - Annotating data for supervised learning is time consuming, expensive and not always practical
  - We can use clustering to group similar data points for sampling
  - It provides a technique for describing the data and possibly getting insights (such as discovering subgroups within a known class, etc.)
- Basic clustering algorithms include **K-Means clustering, hierarchical clustering, and the DBSCAN algorithm**

What is clustering?

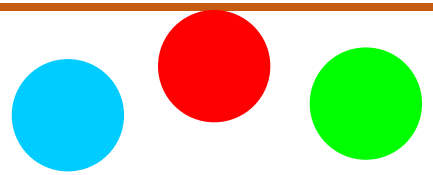
Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



Tan, Steinbach and Kumar, “Introduction to Data Mining”

Types of clusters

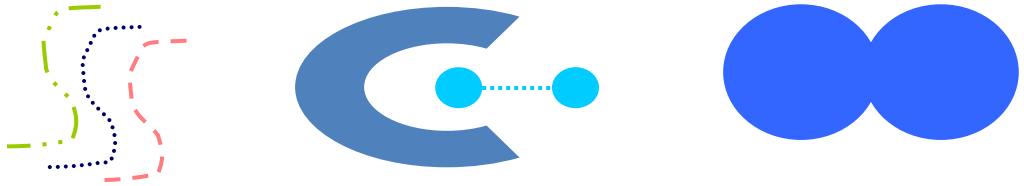
- Well-separated clusters



- Center-based clusters

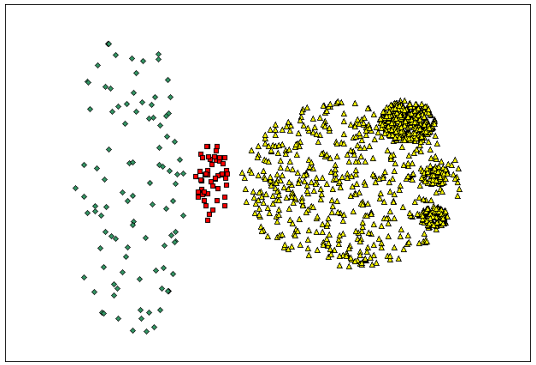
Mean or medioid is the ‘prototype’

- Contiguous clusters



NN or transitive

- Density-based clusters



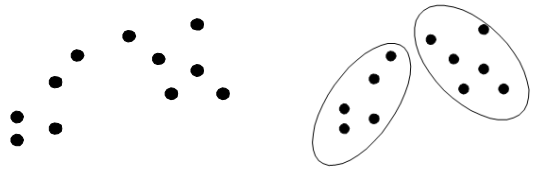
- Property or Conceptual
- Described by an Objective Function

Minimize the edge weight between clusters
and maximize the edge weight within clusters

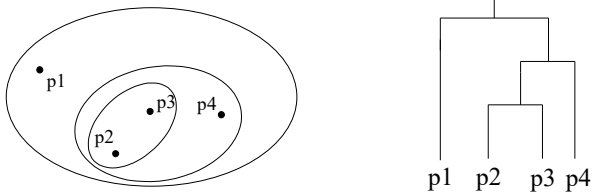
Tan, Steinbach and Kumar, “Introduction to Data Mining”

Types of clustering techniques

- Partitional



- Hierarchical



- Exclusive (vs nonexclusive) – points can belong to multiple clusters
- Fuzzy vs nonfuzzy – points belong to every cluster with a weight in (0,1)
- Property or Conceptual - we only want to cluster some of the data
- Described by an Objective Function – cluster of widely different sizes, shapes, and densities

DATA ANALYTICS

k-means clustering

- Partitional clustering approach
- 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 basic algorithm is very simple

1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change

Tan, Steinbach and Kumar, "Introduction to Data Mining"

k-means parameters

- Initial centroids are often chosen randomly.
  - Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster.
- ‘Closeness’ is measured by Euclidean distance, cosine similarity, correlation, etc.
- K-means will converge for common similarity measures mentioned above.
- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to ‘Until relatively few points change clusters’
- Complexity is O(n * K * I * d)
  - n = number of points, K = number of clusters,
    I = number of iterations, d = number of attributes

Example

One dimensional

Apply K-means algorithm in given data for k=3. Use $C_1(2)$, $C_2(16)$ and $C_3(38)$ as initial cluster centers. Data: 2, 4, 6, 3, 31,12,15,16, 38, 35, 14, 21, 23, 25, 30

Example

One dimensional

Apply K-means algorithm in given data for k=3. Use $C_1(2)$, $C_2(16)$ and $C_3(38)$ as initial cluster centers. Data: 2, 4, 6, 3, 31,12,15,16, 38, 35, 14, 21, 23, 25, 30

$C_1(2)$	$C_2(16)$	$C_3(38)$
m1 = 2	m2 = 16	m3 = 38
{2,3,4,6}	{12,14,15,16,21,23,25}	{31,35,38}

Example

One dimensional

Apply K-means algorithm in given data for k=3. Use $C_1(2)$, $C_2(16)$ and $C_3(38)$ as initial cluster centers. Data: 2, 4, 6, 3, 31,12,15,16, 38, 35, 14, 21, 23, 25, 30

$C_1(2)$	$C_2(16)$	$C_3(38)$
m1 = 2	m2 = 16	m3 = 38
{2,3,4,6}	{12,14,15,16,21,23,25}	{31,35,38}
m1 = 3.75	m2 = 18	m3 = 34.67
{2,3,4,6}	{12,14,15,16,21,23,25}	{31,35,38}

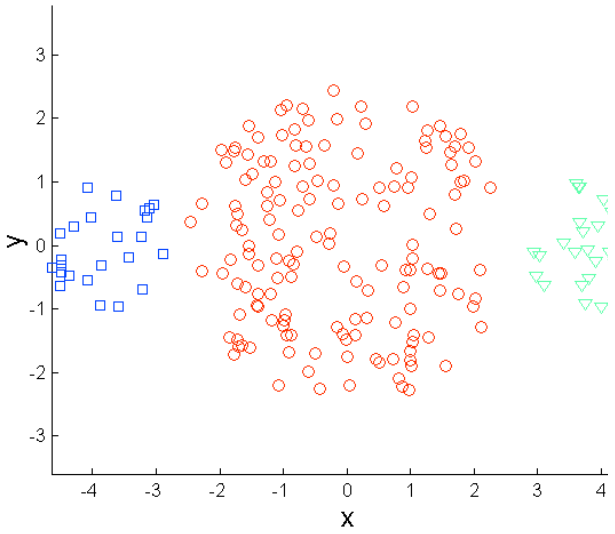
Evaluating k-means clusters

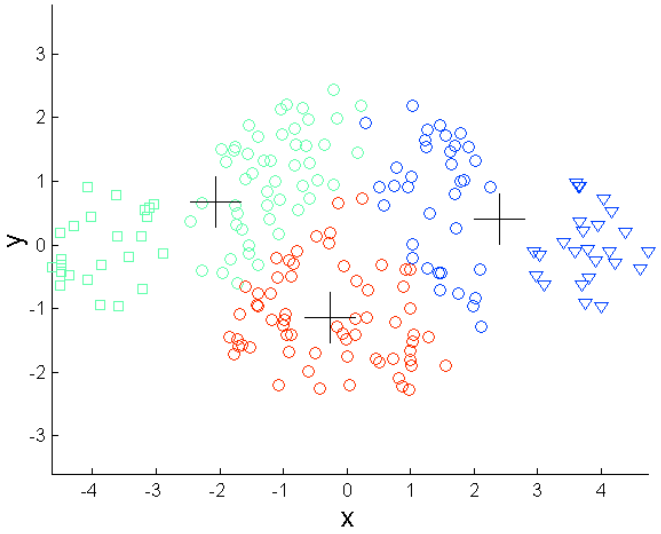
- Most common measure is Sum of Squared Error (SSE)
  - For each point, the error is the distance to the nearest cluster
  - To get SSE, we square these errors and sum them.

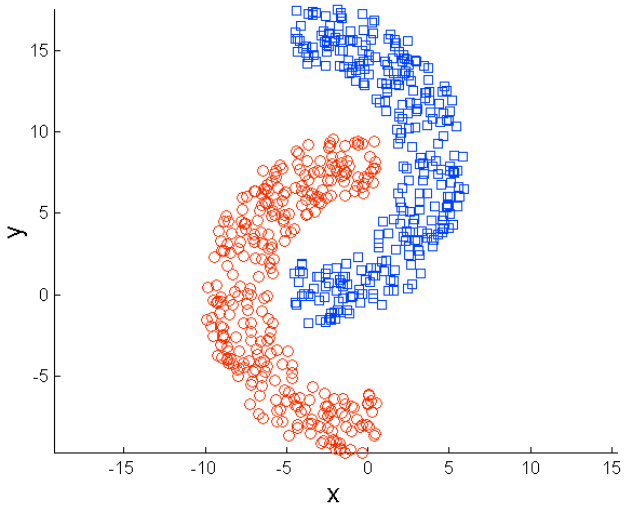
$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

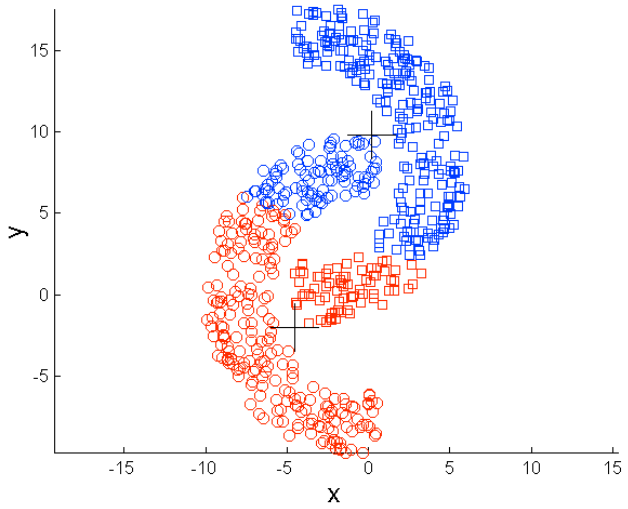
  - x is a data point in cluster C_i and m_i is the representative point for cluster C_i
    - can show that m_i corresponds to the center (mean) of the cluster
  - Given two clusters, we can choose the one with the smallest error
  - One easy way to reduce SSE is to increase K, the number of clusters
    - A good clustering with smaller K can have a lower SSE than a poor clustering with higher K

k-means clustering – some cons



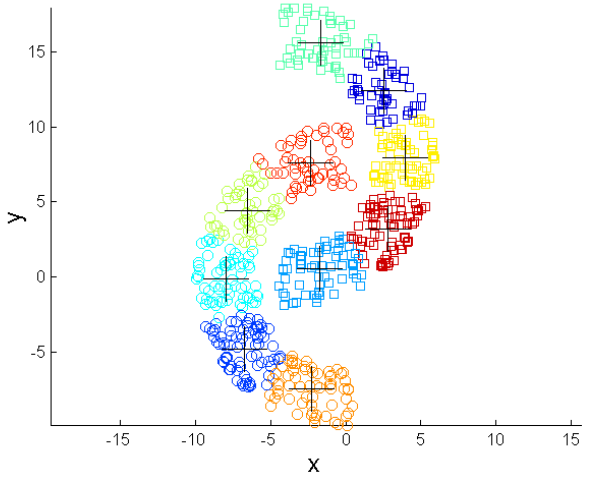






- Different configurations for each run due to random initialization
- Works well only for similarly shaped (or sized clusters)
- Does not work well for inherently nonglobular clusters

Try: (1) Choosing k>> no. of clusters
(2) Run kmeans multiple times; select the best configuration



Tan, Steinbach and Kumar, “Introduction to Data Mining”

k-means clustering – some cons

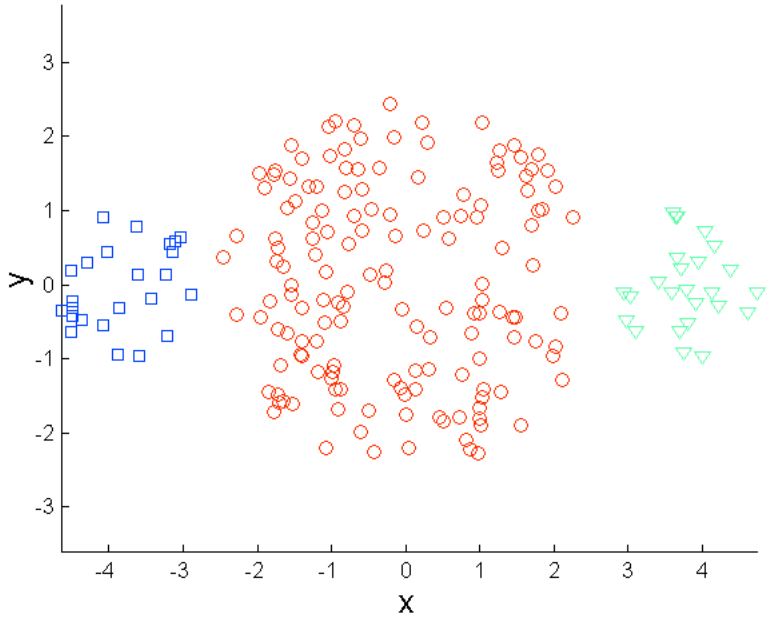
K-means has problems when clusters are of differing

- Sizes
- Densities
- Non-globular shapes

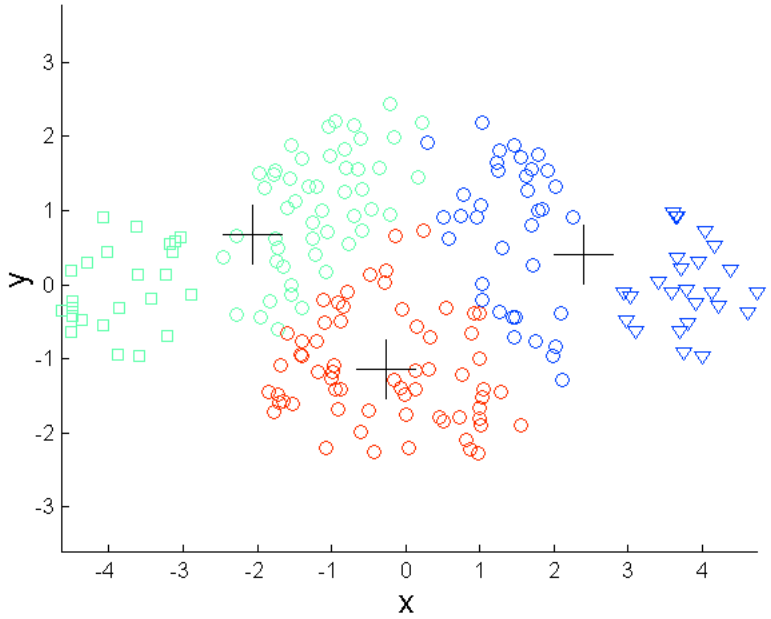
K-means has problems when the data contains outliers.

k-means clustering – some cons

Limitations of K-means: Differing Sizes



Original Points

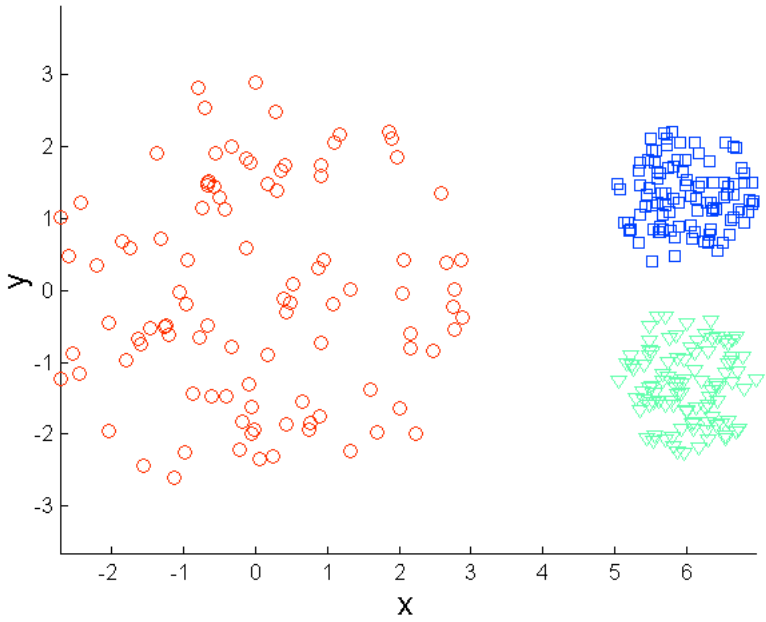


K-means (3 Clusters)

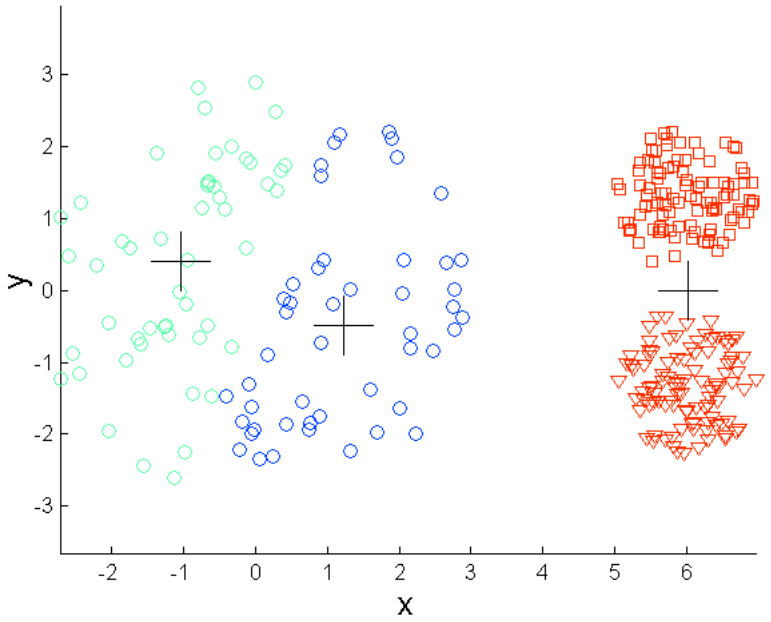
Tan, Steinbach and Kumar, "Introduction to Data Mining"

k-means clustering – some cons

Limitations of K-means: Differing Density



Original Points

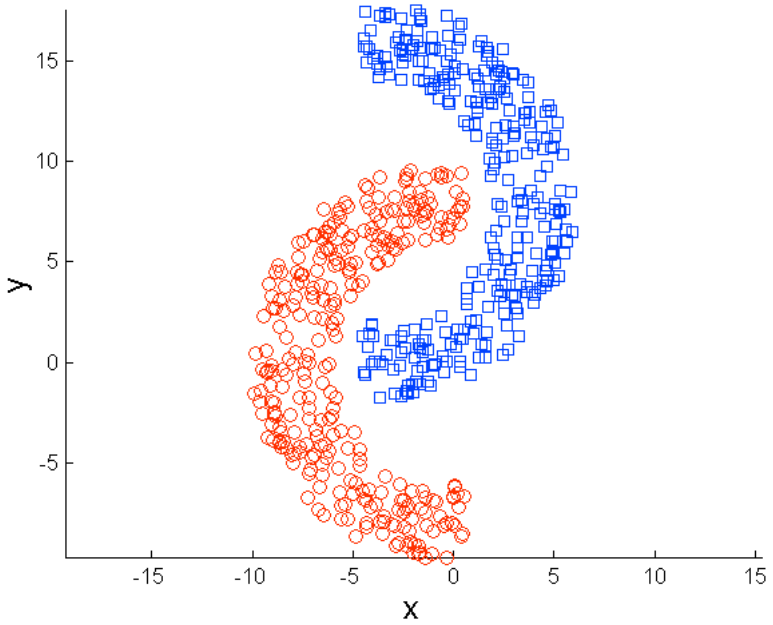


K-means (3 Clusters)

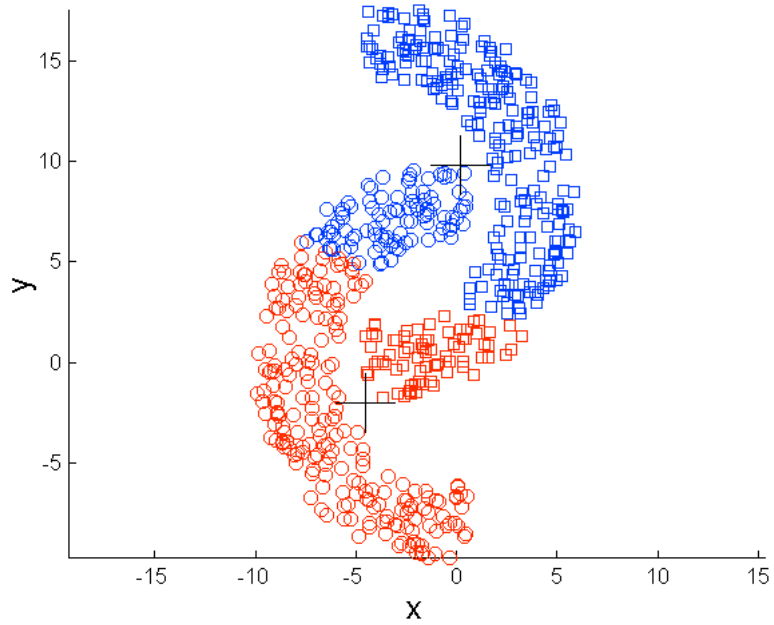
Tan, Steinbach and Kumar, “Introduction to Data Mining”

k-means clustering – some cons

Limitations of K-means: Non-globular Shapes



Original Points

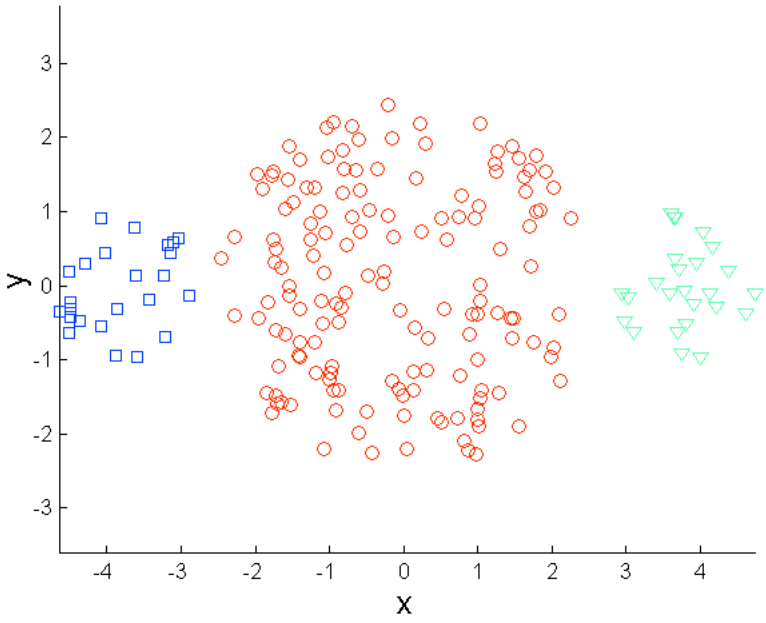


K-means (2 Clusters)

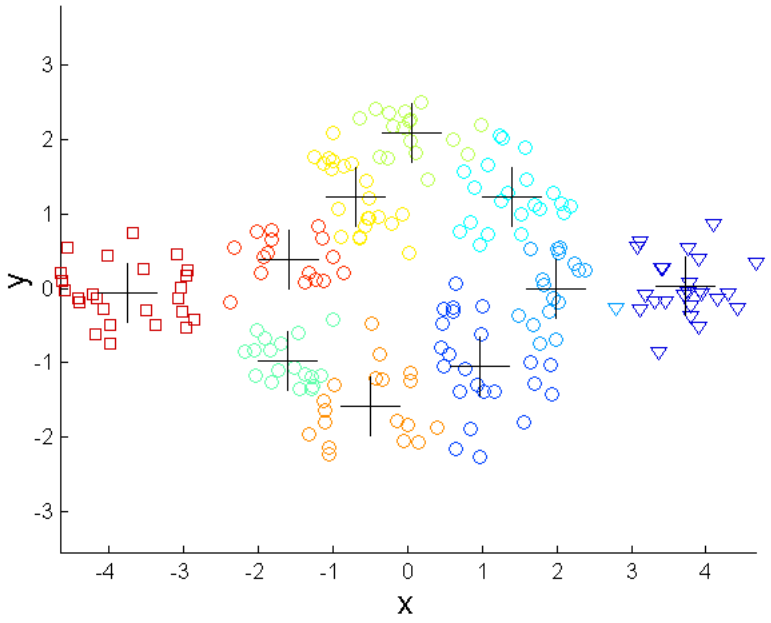
Tan, Steinbach and Kumar, “Introduction to Data Mining”

k-means clustering – some cons

Overcoming K-means Limitations



Original Points



K-means Clusters

One solution is to use many clusters.

Find parts of clusters, but need to put together.

Tan, Steinbach and Kumar, “Introduction to Data Mining”

Quality and optimal number of clusters

- Number of clusters and recommended index (Calinski and Harabasz)

$$CH(k) = \frac{B(k)/k-1}{W(k)/(n-k)}$$

- Hartigan statistic

$$H(k) = \{(W(k)/W(k-1)) - 1\}/(n-k-1)$$

- Silhouette statistic

$$S(i) = (b(i)-a(i)/max\{a(i),b\{i\}))$$

Which can be also written as:

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i) \end{cases}$$

From the above definition it is clear that

$$-1 \leq s(i) \leq 1$$

For data point $i \in C_i$ (data point i in the cluster C_i), let

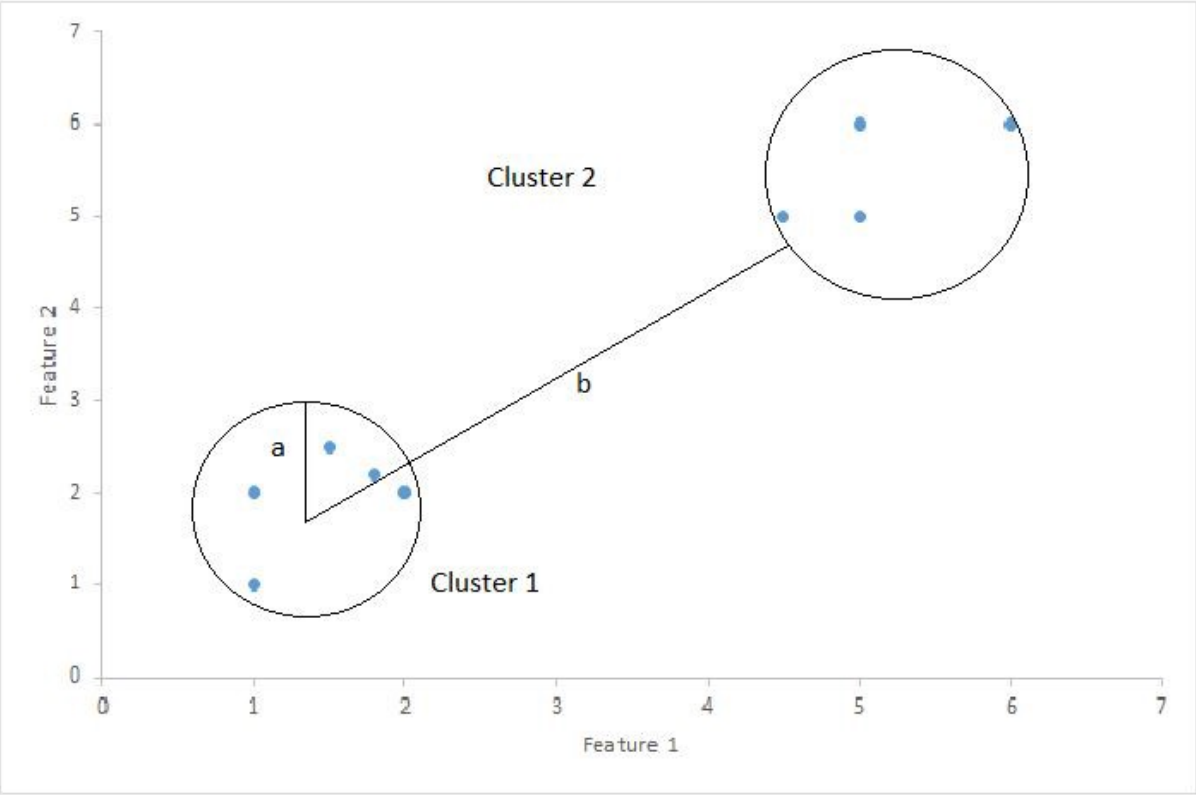
$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i,j) \qquad b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i,j)$$

Tan, Steinbach and Kumar, "Introduction to Data Mining"

Quality and optimal number of clusters

Silhouette statistic

$$S(i) = (b(i)-a(i)/max\{a(i),b\{i\}\})$$

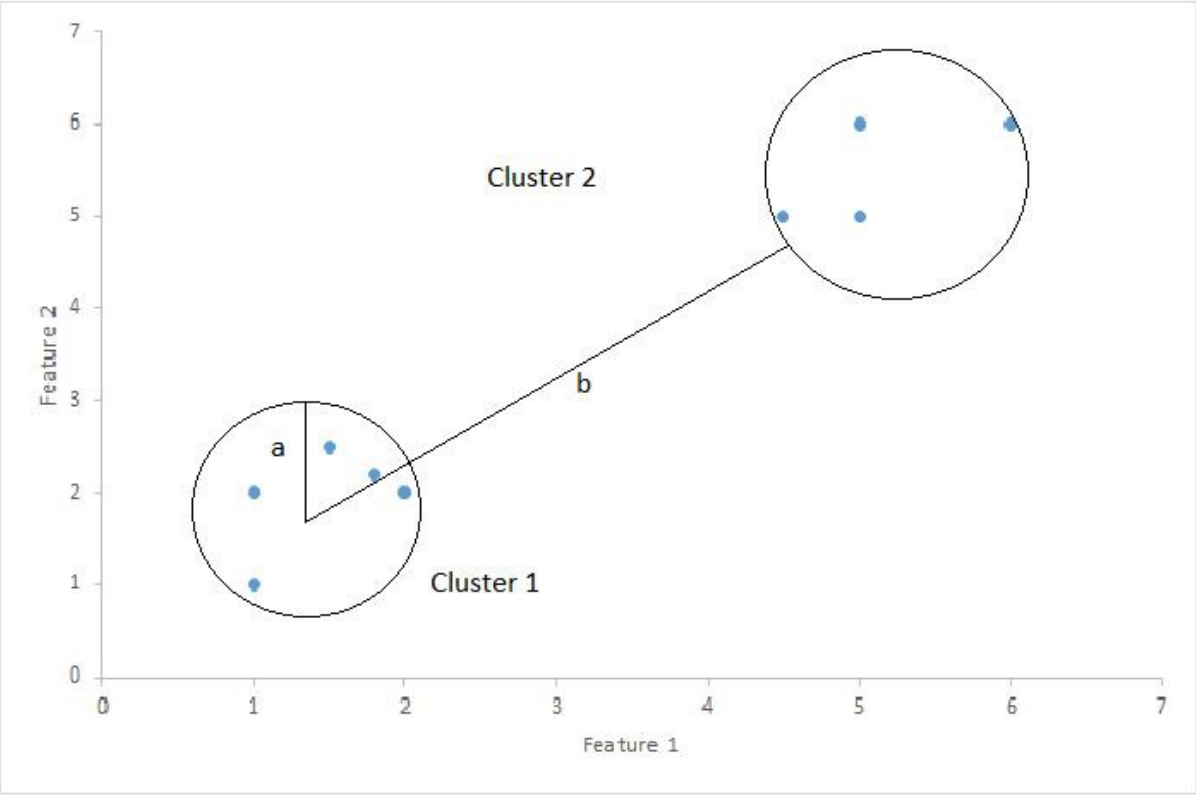


https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c

Quality and optimal number of clusters

Silhouette statistic

$S(i) = (b(i)-a(i)/max\{a(i),b\{i\}\})$



Silhouette Coefficient:

Silhouette Coefficient or silhouette score is a metric used to calculate the goodness of a clustering technique. Its value ranges from -1 to 1.

1: Means clusters are well apart from each other and clearly distinguished.

0: Means clusters are indifferent, or we can say that the distance between clusters is not significant.

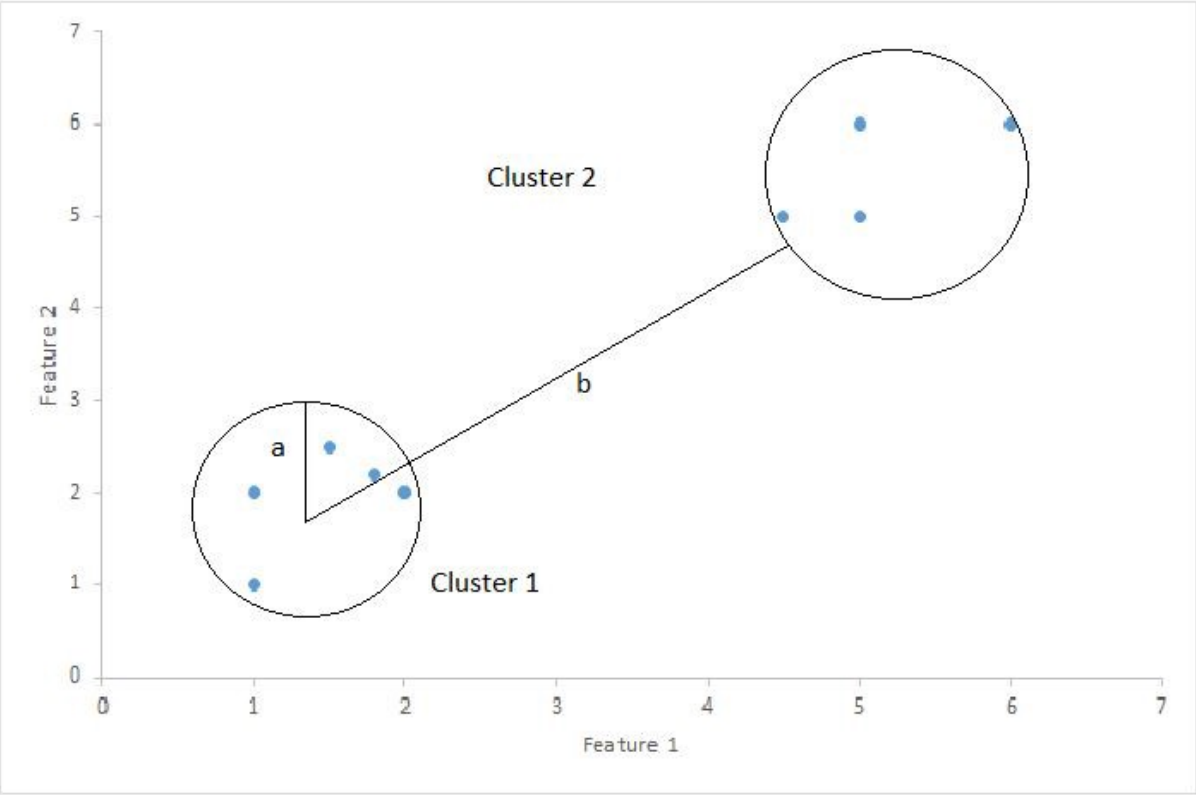
-1: Means clusters are assigned in the wrong way.

https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c

Quality and optimal number of clusters

Silhouette statistic

$S(i) = (b(i)-a(i)/max\{a(i),b\{i\}\})$



Silhouette Score = (b-a)/max(a,b)

where

a= average intra-cluster distance i.e the average distance between each point within a cluster.

b= average inter-cluster distance i.e the average distance between all clusters.

https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c

Quality and optimal number of clusters

To evaluate which number of clusters is more optimum for our dataset, or find *cluster fitness* we use two scoring methods — **Silhouette Coefficient** and **Calinski Harabasz Score.**

The Silhouette Coefficient is calculated using the **mean intra-cluster distance (a)** and the **mean nearest-cluster distance (b)** for each sample. The Silhouette Coefficient for a sample is **(b-a) / max(b-a)**

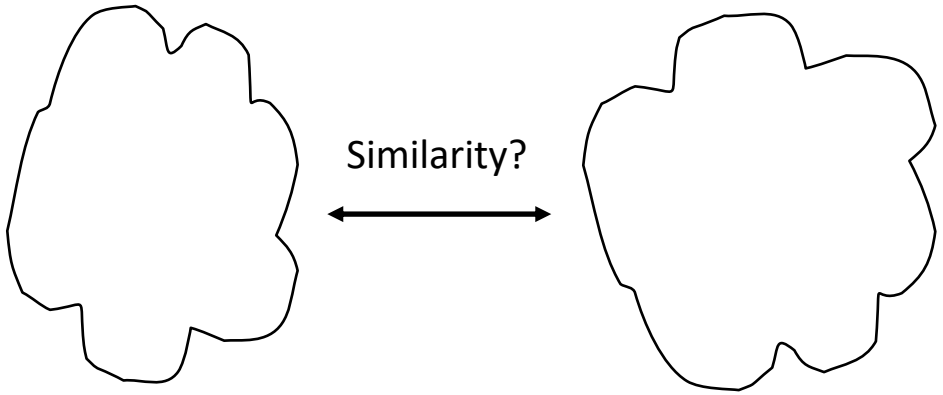
The Calinski Harabasz Score or Variance Ratio is the ratio between **within-cluster dispersion** and **between-cluster dispersion**

no. of clusters	silhouette coefficient	ch score
5	0.26	48068.3
8	0.24	41104.9
12	0.22	36554.9

Both the values are higher than they were for our earlier clusters 12 and 8. We can conclude that k=5 is our optimal number of clusters.

https://towardsdatascience.com/clustering-with-k-means-1e07a8bfb7ca

Agglomerative hierarchical clustering



	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

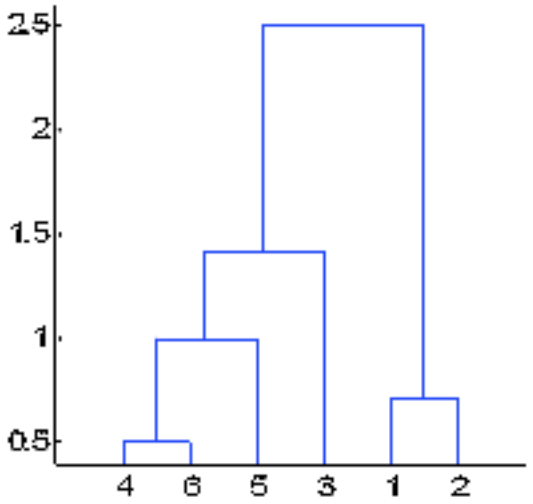
Agglomerative hierarchical clustering

Dendrogram

The standard output of hierarchical clustering is a dendrogram.

A dendrogram is a cluster tree diagram where the distance of split or merge is recorded.

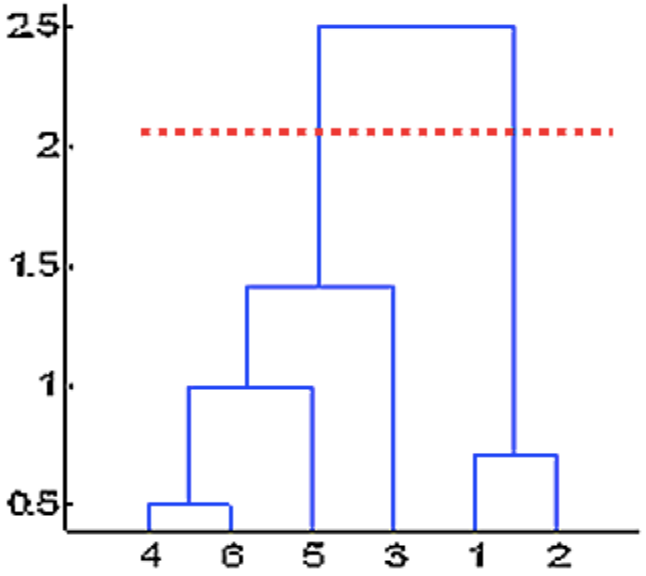
Dendrogram is a visualization of hierarchical clustering.



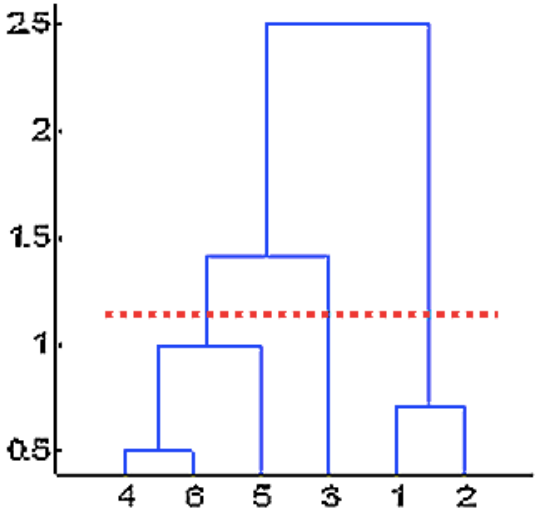
Agglomerative hierarchical clustering

Using dendrogram, we can easily specify the cutting point to determine number of clusters.

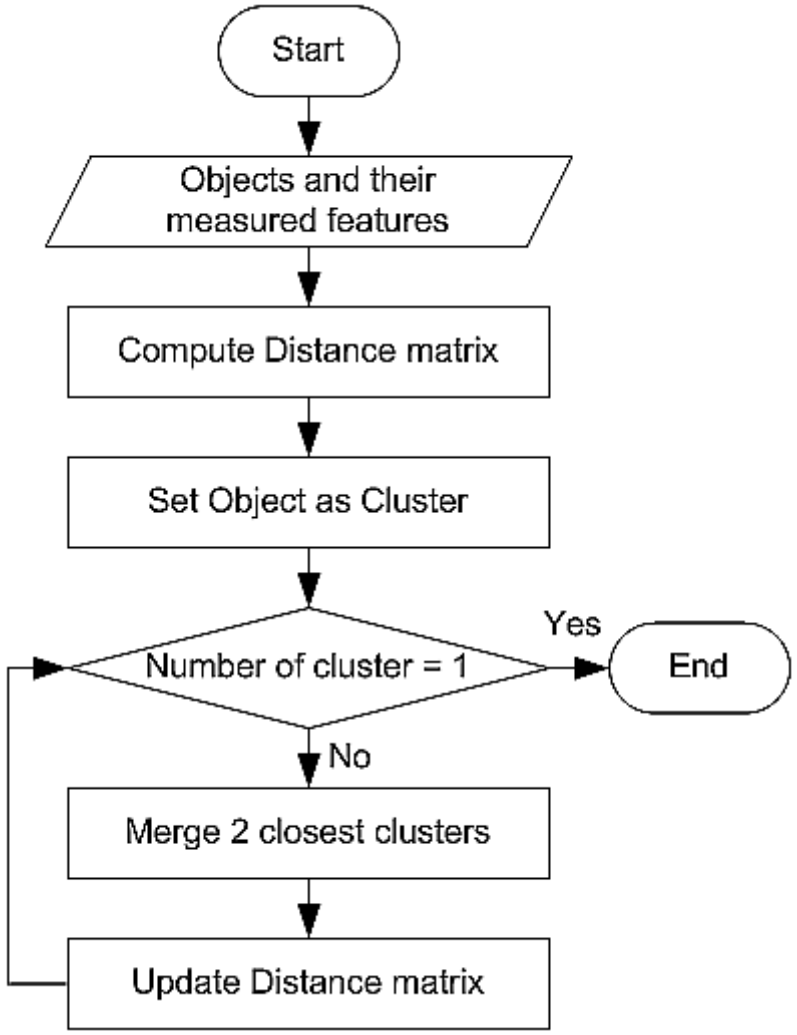
For example, in the top dendrogram below, we set cutting distance at 2 and we obtain two clusters out of 6 objects. The first cluster consists of 4 objects (number 4, 6, 5 and 3) and the second cluster consists of two objects (number 1 and 2).



Similarly, in the down dendrogram, setting cutting distance at 1.2 will produce 3 clusters.



Agglomerative hierarchical clustering



Agglomerative hierarchical clustering

Agglomerative hierarchical clustering

References

Text Book:

"Business Analytics, The Science of Data-Driven Making", U. Dinesh Kumar, Wiley 2017 (Chapter 14.1-14.2.6, 14.3-14.6)

"Recommender Systems, The text book, Charu C. Aggarwal, Springer 2016 Section 1.and Section 2.

Image Courtesy

https://www.analyticsvidhya.com/blog/2020/09/how-dbscan-clustering-works/

THANK YOU

Jyothi R.
Assistant Professor,
Department of Computer Science
jyothir@pes.edu