Unsupervised Machine Learning

Learning Objectives

Upon completing this assignment, students will:

- Learn about unsupervised machine learning
- Recognize clustering as an unsupervised machine learning task
- Become familiar with how the k-means clustering algorithm works

Road map

- **Basic concepts**
- **K-means algorithm**
- **Distance functions**

Supervised vs. unsupervised learning

- Supervised learning: learn models or classifiers from the data that relate data attributes to a target class attribute.
  - These models are then used to predict the values of the class attribute in test or future data instances.
- Unsupervised learning: The data have no target/class attribute.
  - We want to explore the data to find some intrinsic structures in them.

Why Use Unsupervised Machine Learning?

- Because of the ability to uncover previously overlooked patterns in complex datasets, unsupervised machine learning has many applications, and is used in fields.

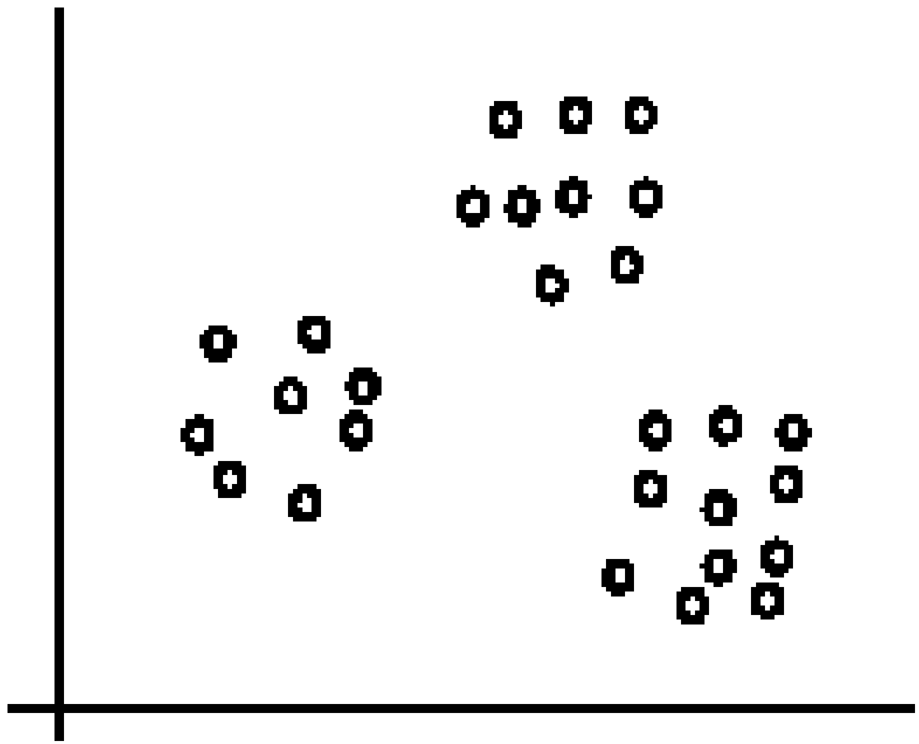
- Unsupervised learning can be used to model and organize large quantities of unstructured data—uncovering some intrinsic structure directly from the data itself.

Clustering

- Clustering is one main approach to unsupervised learning.
  - It finds similarity groups in data, called **clusters,** it groups data instances that are similar to (near) each other in one cluster and data instances that are very different (far away) from each other into different clusters.

An illustration

- The data set has three natural groups of data points, i.e., 3 natural clusters.



What is clustering for?

- Let us see some real-life examples
- Example 1: group people of similar sizes together to make "small", "medium" and "large" T-Shirts.
  - Tailor-made for each person: too expensive
  - One-size-fits-all: clearly bad
- Example 2: In marketing, segment customers according to their similarities
  - To do targeted marketing.

Aspects of clustering

- A clustering algorithm
  - Partitional clustering
- A distance (similarity, or dissimilarity) function
- Clustering quality
  - Inter-clusters distance $\Rightarrow$ maximized
  - Intra-clusters distance $\Rightarrow$ minimized
- The quality of a clustering result depends on the algorithm, the distance function, and the application.

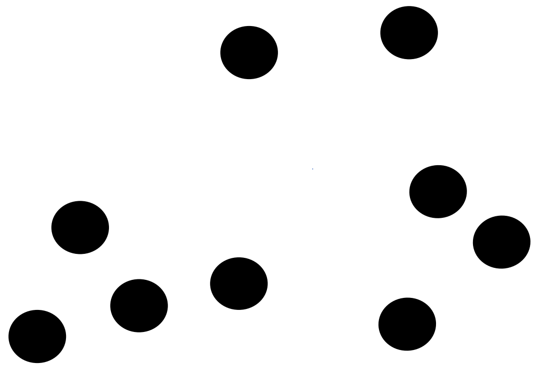
Road map

- **Basic concepts**
- **K-means algorithm**
- **Distance functions**

K-Means Clustering Process

- The input is: (1) a set of data; (2) a number of clusters that you specify, known as k.

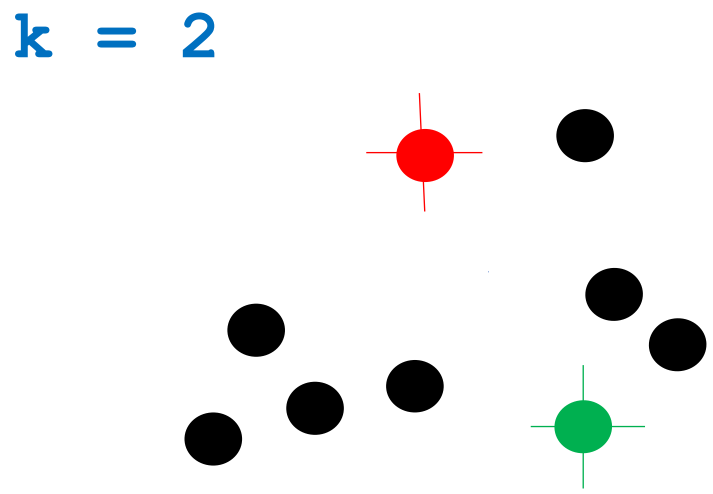
`k = 2`



K-Means Clustering Process (cont.)

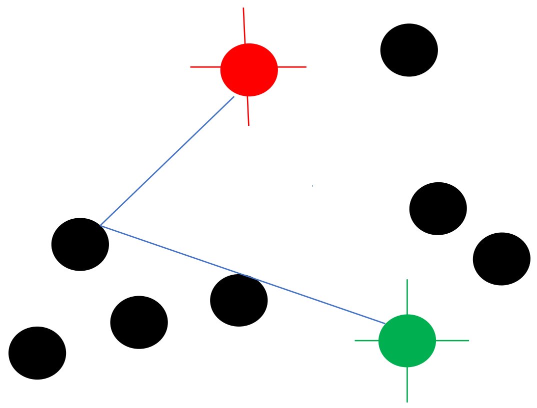
- Each cluster has one centroid. The centroid (also sometimes called a seed) is the cluster's center.

- The algorithm selects k data points as initial centroids (either randomly, or by a more informed method).



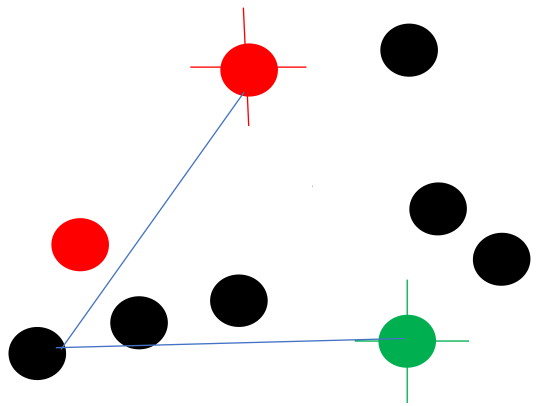
K-Means Clustering Process (cont.)

- The distance between a data point is measured to each of the two centroids.



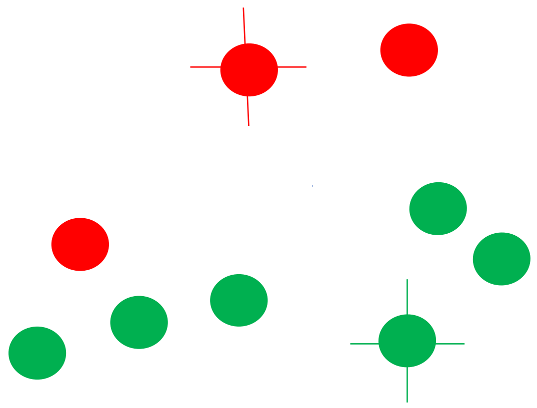
K-Means Clustering Process (cont.)

- That data point is then assigned to the cluster of the closest centroid (indicated here by changing to red).
- The algorithm calculates the distance between the next data point and the k centroids.



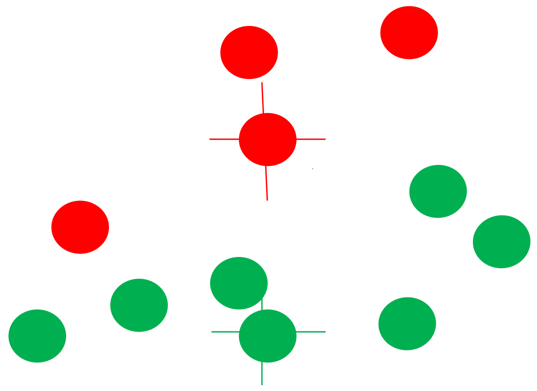
K-Means Clustering Process (cont.)

- This continues for each of the data points until they are each assigned to the cluster of the nearest centroid.



K-Means Clustering Process (cont.)

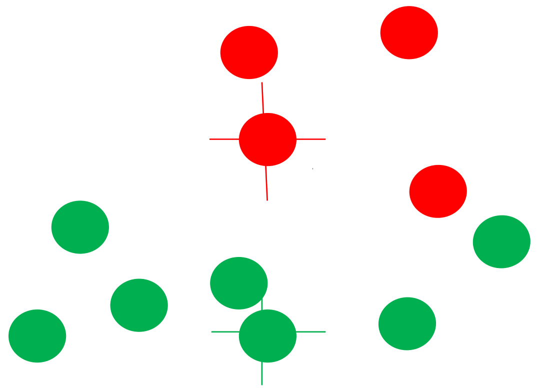
- Centroids are now recomputed. The new centroids are created by taking the mean, or average, of all the data points assigned to the cluster.



- Note that some of the data points may now be closer to one of the new centroids.

K-Means Clustering Process (cont.)

- The process of measuring the distance to each centroid and reassigning datapoints to the nearest centroid continues iteratively.

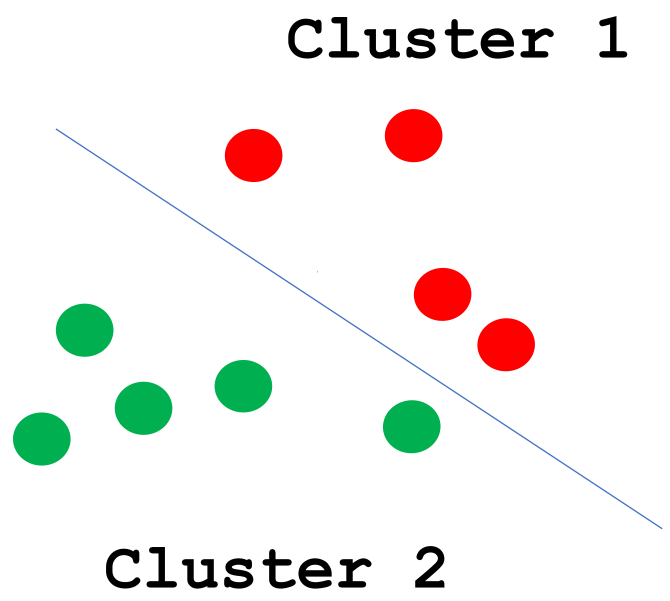


- Likewise, each centroid continues to be moved to the cluster's mean in this iterative process.

K-Means Clustering Process (cont.)

- Finally, stopping criteria are met (e.g., no data points change clusters).

- K-means converges, to a local—but not necessarily global minimum—meaning that different iterations, with different initial centroids, can produce differing results.



Summary: K-Means Clustering

`k = number of clusters`

1. Select k initial cluster centroids

2. Assign data points to nearest centroid

3. Recalculate centroids as the mean of each cluster of data points

4. Repeat steps 2 and 3 until convergence

A disk version of *k*-means

- K-means can be implemented with data on disk
  - In each iteration, it scans the data once.
  - as the centroids can be computed incrementally
- It can be used to cluster large datasets that do not fit in main memory
- We need to control the number of iterations
  - In practice, a limited is set (< 50).
- Not the best method. There are other scale-up algorithms, e.g., BIRCH.

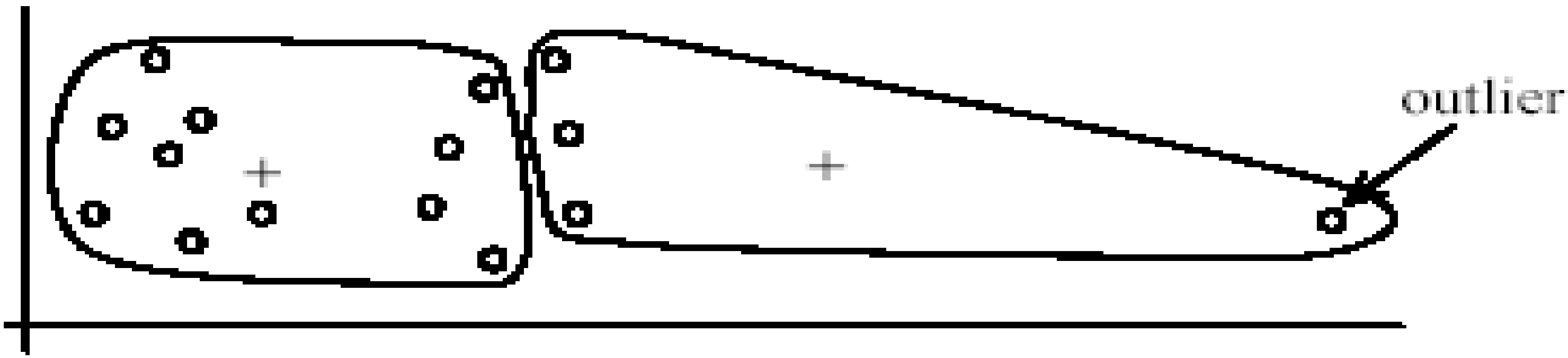
Strengths of k-means

- Strengths:
  - Simple: easy to understand and to implement
  - Efficient: Time complexity depends on (*tkn*), where *n* is the number of data points, *k* is the number of clusters, and *t* is the number of iterations.
- K-means is the most popular clustering algorithm.
- Note that: it terminates at a local optimum. The global optimum is hard to find due to complexity.

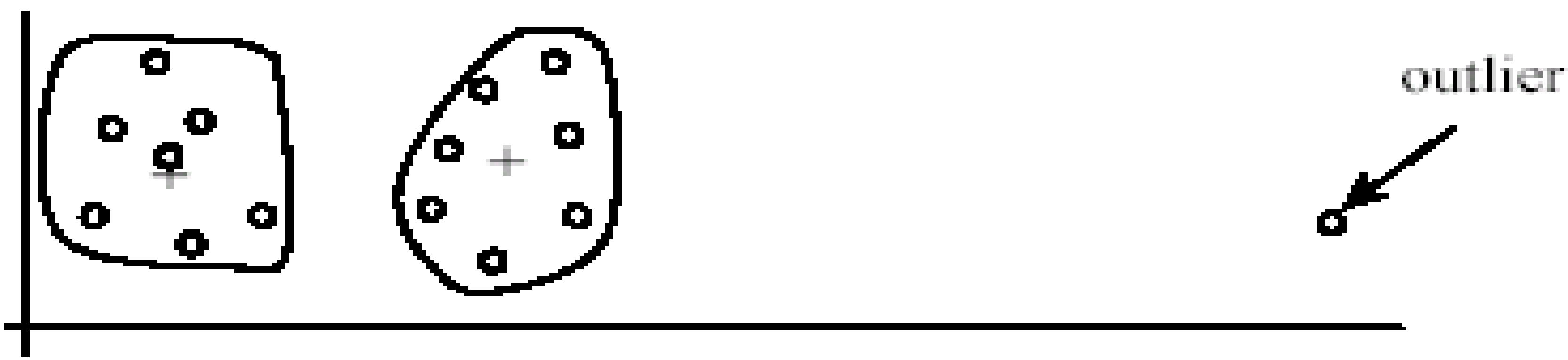
Weaknesses of k-means

- The algorithm is only applicable if the mean is defined.
  - For categorical data, *k*-mode - the centroid is represented by the most frequent values.
- The user needs to specify *k*.
- The algorithm is sensitive to **outliers**
  - Outliers are data points that are very far away from other data points.
  - Outliers could be errors in the data recording or some special data points with very different values.

Weaknesses of k-means: Problems with outliers



(A): Undesirable clusters



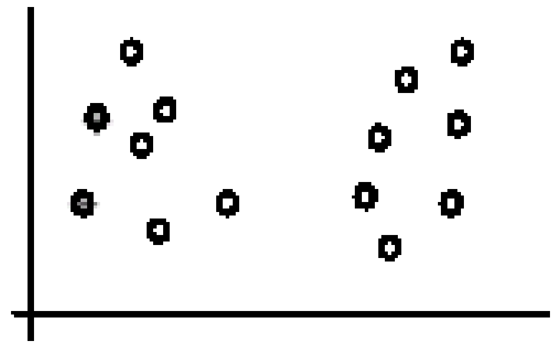
(B): Ideal clusters

Weaknesses of k-means: To deal with outliers

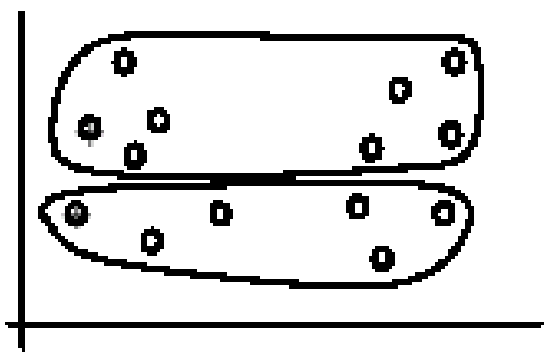
- One method is to remove some data points in the clustering process that are much further away from the centroids than other data points.
- Another method is to perform random sampling. Since in sampling we only choose a small subset of the data points, the chance of selecting an outlier is very small.

Weaknesses of k-means (cont …)

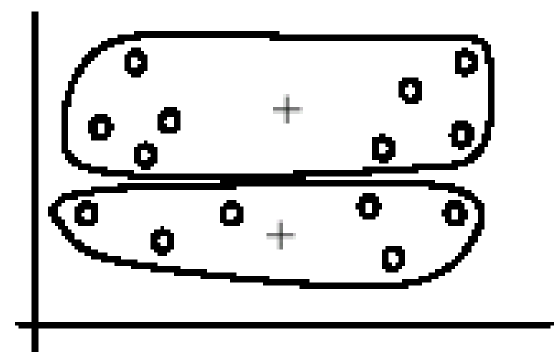
- The algorithm is sensitive to initial seeds.



(A). Random selection of seeds (centroids)



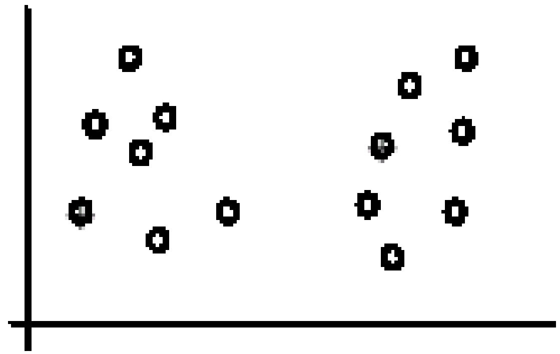
(B). Iteration 1



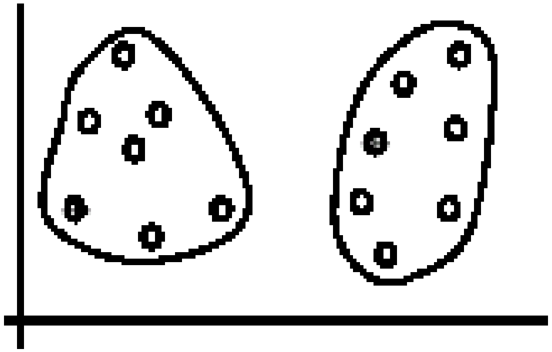
(C). Iteration 2

Weaknesses of k-means (cont ...)

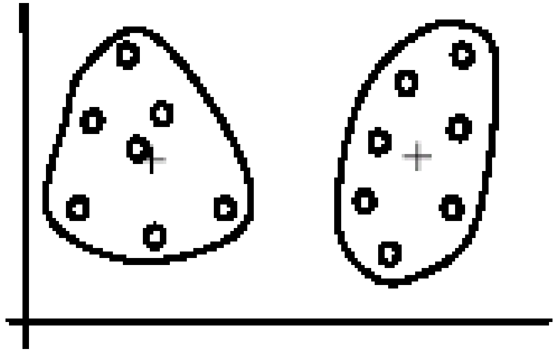
- If we use different seeds: good results



(A). Random selection of *k* seeds (centroids)



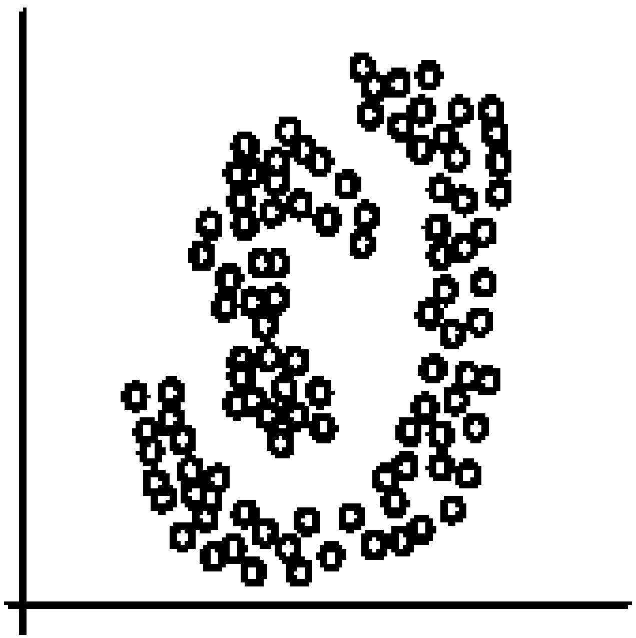
(B). Iteration 1



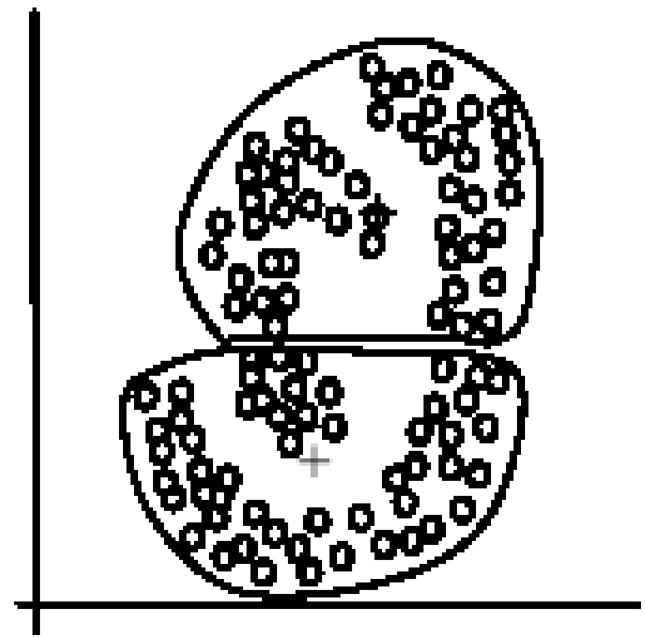
(C). Iteration 2

Weaknesses of k-means (cont …)

- The *k*-means algorithm is not suitable for discovering clusters that are not hyper-ellipsoids (or hyper-spheres).



(A): Two natural clusters



(B): *k*-means clusters

K-means summary

- Despite weaknesses, *k*-means is still the most popular algorithm due to its simplicity, efficiency
  - other clustering algorithms have their own lists of weaknesses.
  - No clear evidence that any other clustering algorithm performs better in general
    - although they may be more suitable for some specific types of data or applications.
- Comparing different clustering algorithms is a difficult task. No one knows the correct clusters

Road map

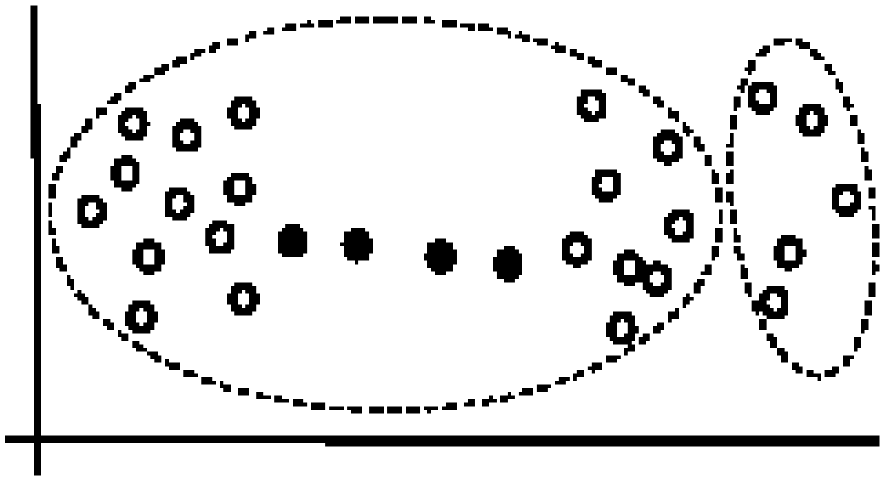
- **Basic concepts**
- **K-means algorithm**
- **Distance functions**

Measuring the distance of two clusters

- A few ways to measure distances of two clusters.
- Results in different variations of the algorithm.
  - Single link
  - Complete link
  - Average link
  - Centroids

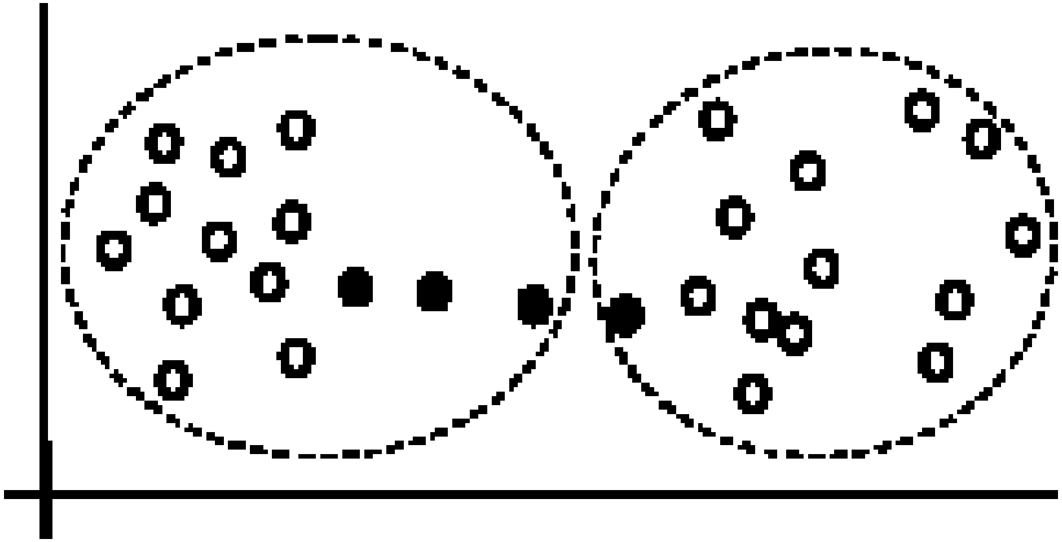
Single link method

- The distance between two clusters is the distance between two closest data points in the two clusters, one data point from each cluster.



Complete link method

- The distance between two clusters is the distance of two furthest data points in the two clusters.
- It is sensitive to outliers because they are far away



Average link and centroid methods

- Average link: In this method, the distance between two clusters is the average distance of all pair-wise distances between the data points in the two clusters.
- Centroid method: In this method, the distance between two clusters is the distance between their centroids

Distance functions

- Key to clustering. "similarity" and "dissimilarity" can also commonly used terms.
- There are numerous distance functions for
  - Different types of data
    - Numeric data
    - Nominal data
  - Different specific applications

Distance functions for numeric attributes

- Most commonly used functions are
  - Euclidean distance and
  - Manhattan (city block) distance
- We denote distance with: *dist*($\mathbf{x}_i$, $\mathbf{x}_j$), where $\mathbf{x}_i$ and $\mathbf{x}_j$ are two data points (vectors)
- They are special cases of Minkowski distance. h is positive integer.

$$dist(\mathbf{x}_i, \mathbf{x}_j) = ((x_{i1} - x_{j1})^h + (x_{i2} - x_{j2})^h + ... + (x_{ir} - x_{jr})^h)^{\frac{1}{h}}$$

Euclidean distance and Manhattan distance

- If h = 2, it is the Euclidean distance

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \ldots + (x_{ir} - x_{jr})^2}$$

- If h = 1, it is the Manhattan distance

$$dist(\mathbf{x}_i, \mathbf{x}_j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ir} - x_{jr}|$$

- Weighted Euclidean distance

$$dist(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{w_1(x_{i1} - x_{j1})^2 + w_2(x_{i2} - x_{j2})^2 + \ldots + w_r(x_{ir} - x_{jr})^2}$$

Conclusion

- Using unsupervised machine learning with a clustering algorithm results in unlabeled clusters of data based on commonalities.

- Currently, the most popular technique for clustering in unsupervised learning is a partitioning method known as the k-means algorithm.

- In k-means clustering, each cluster has a center point, known as a centroid.

Conclusion (cont.)

- The steps for k-means clustering are as follows:

1. Select k initial cluster centroids

2. Assign data points to nearest centroid

3. Recalculate centroids as the mean of each cluster of data points

4. Repeat steps 2 and 3 until convergence