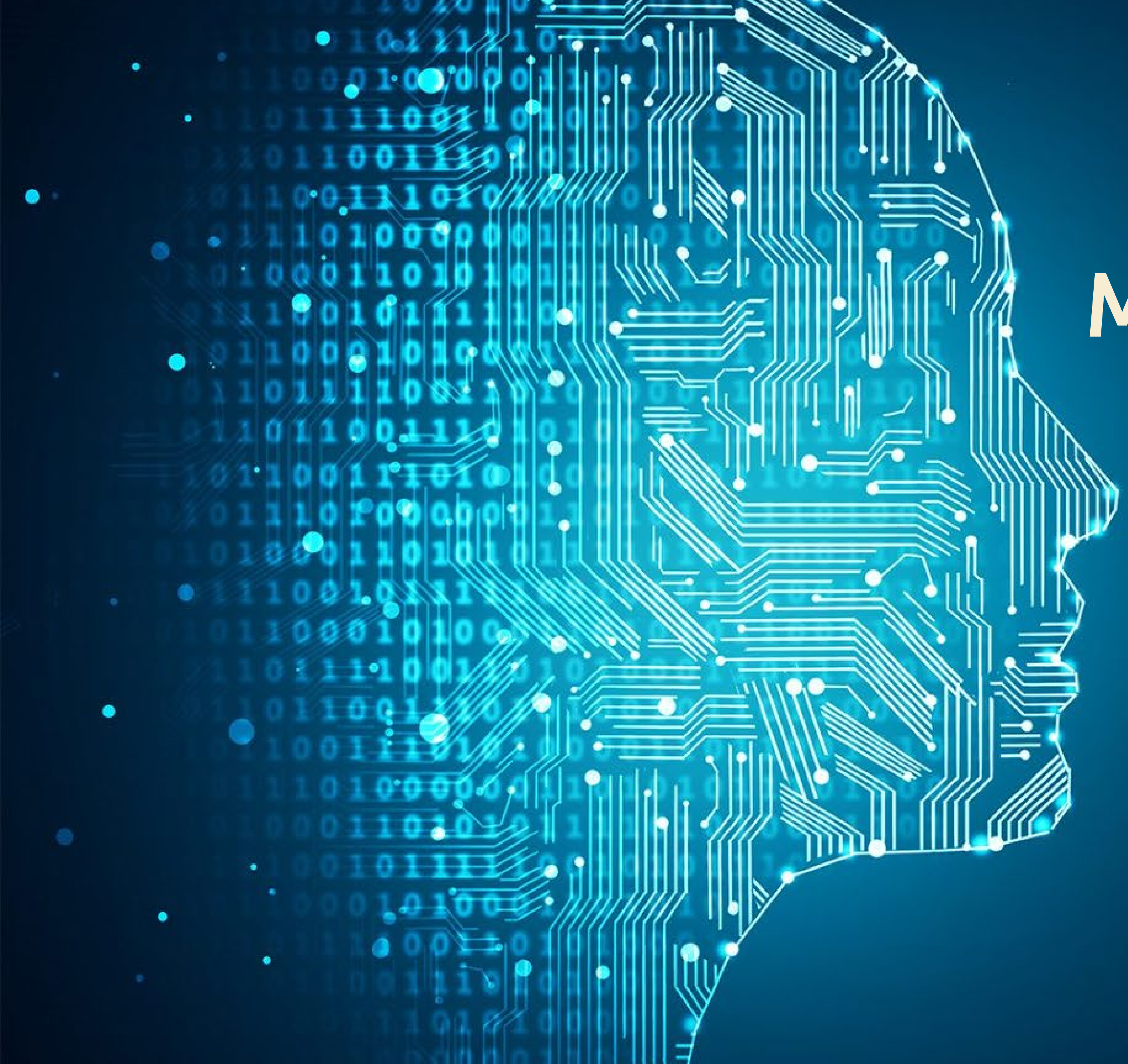


Applied Machine Learning

(BUAN 6341)

Lecture 8
Clustering

Yingjie Zhang
University of Texas at Dallas
yingjie.zhang@utdallas.edu

Spring 2019

Exam I

Statistics:

Mean:	86.56
Median:	88.5
Highest:	99

Agenda

- Overview of Unsupervised Learning
- Clustering

Unsupervised Learning

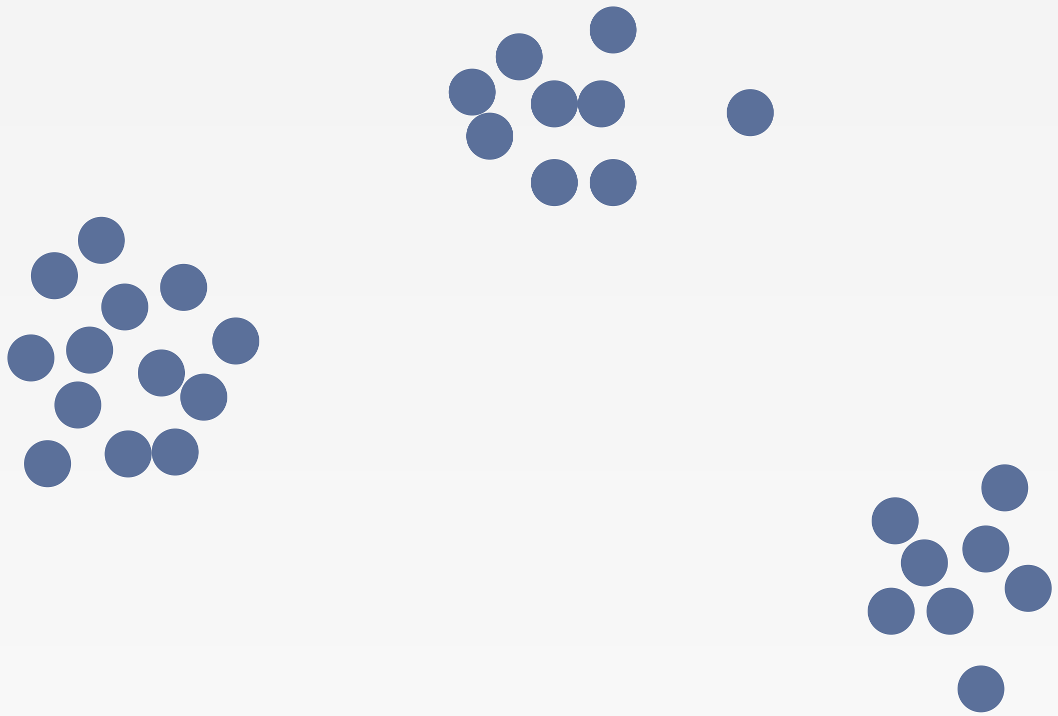
Goals

- To discover interesting things from the data:
  - Is there an informative way to visualize the data?
  - Can we discover subgroups among the variables?

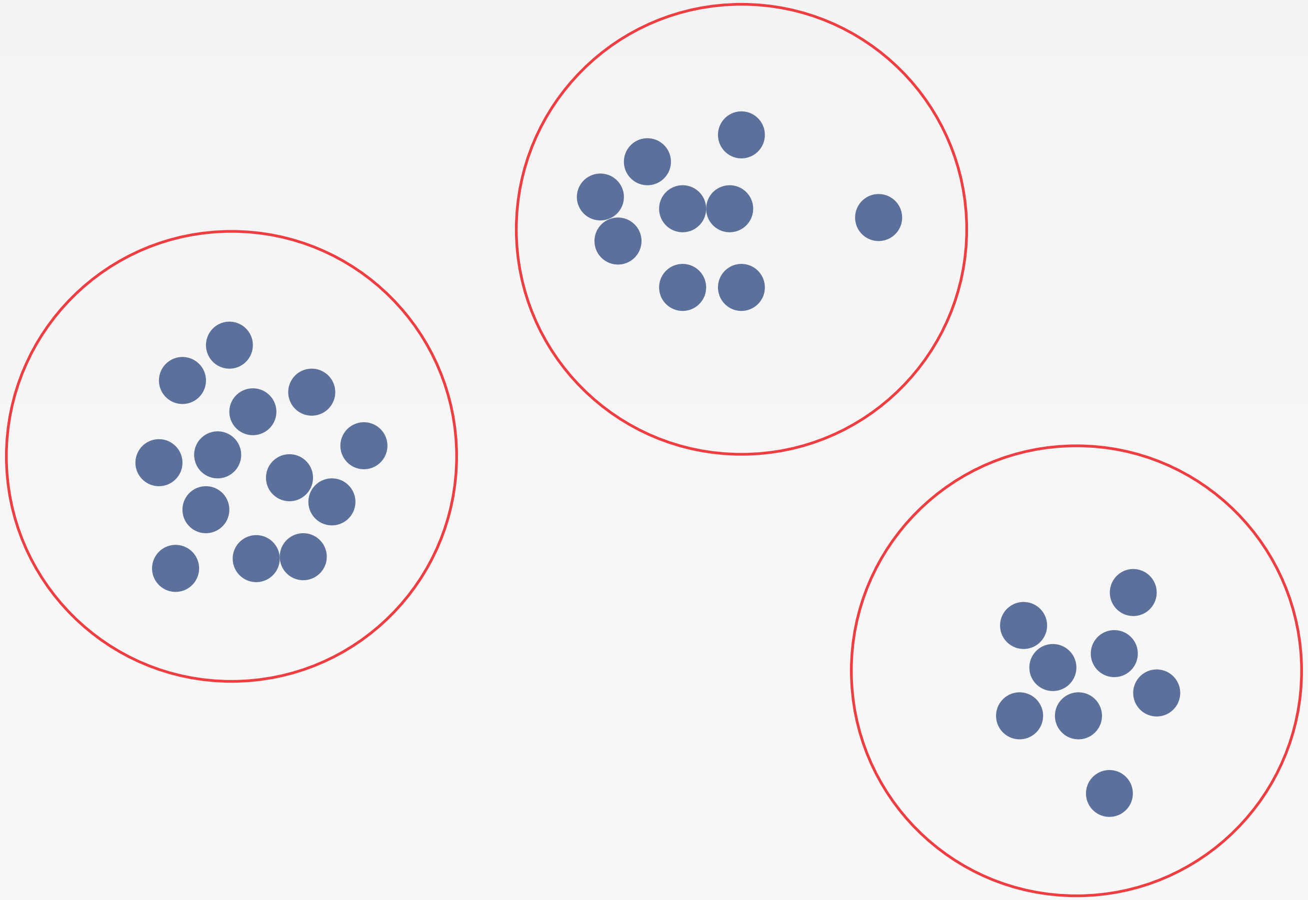
- Models:
  - Clustering
    - K-means
    - DBSCAN
    - Hierarchical Clustering

Clustering

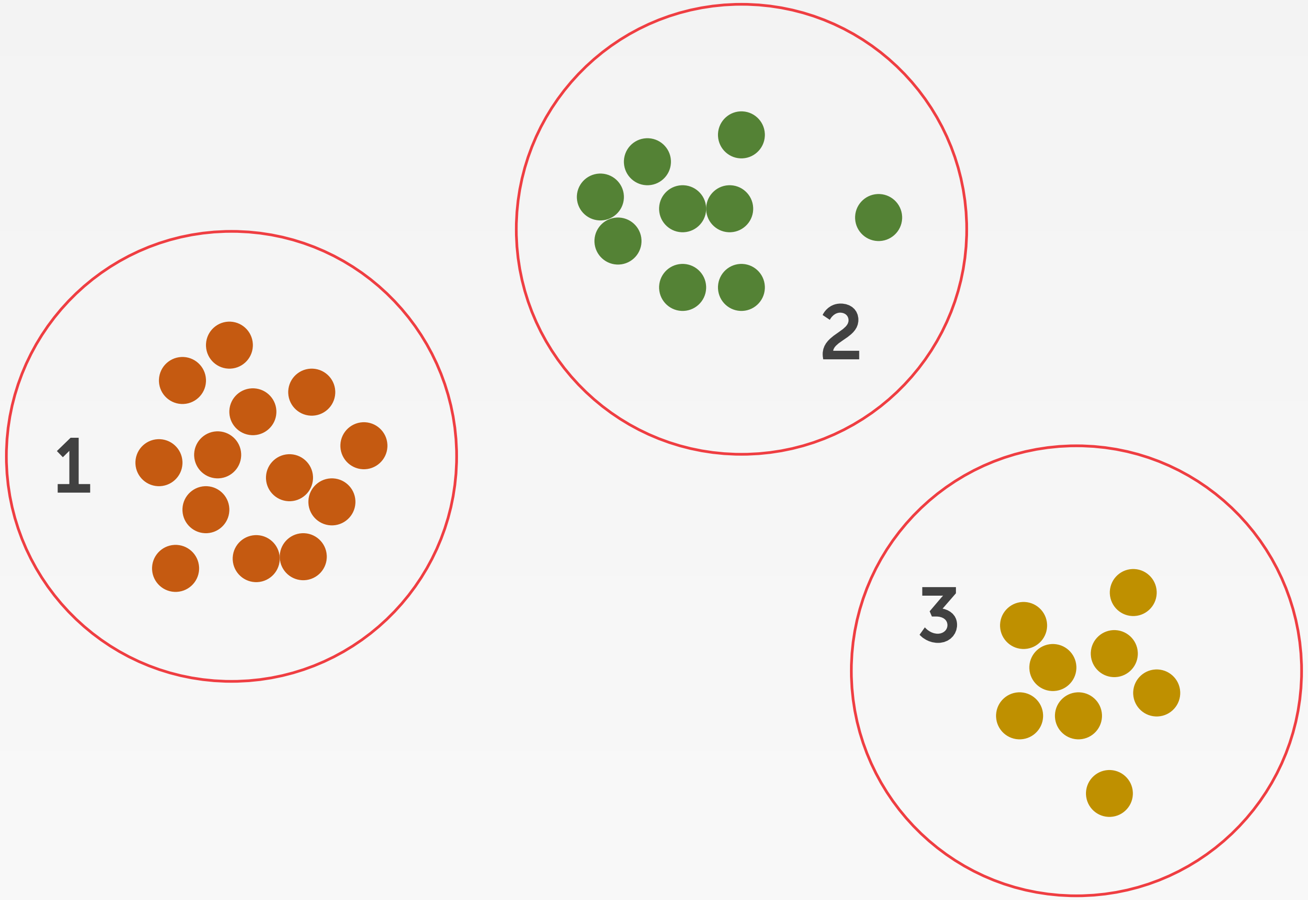
What is Clustering?



What is Clustering?



What is Clustering?



UT DALLAS
Naveen Jindal
School of Management

Clustering

- Partition data into groups (clusters)
- Points within a cluster should be “similar”
- Points in different clusters should be “different”

The Clustering Problem

- We are given a set of data points $X_1, X_2, \dots, X_m$ that we would like to cluster
- Each data point has n-dimensional features: $X = (x_1, x_2, \dots x_n)$
- We do not make any statistical assumption on the given data

K-Means

Overview

- K-means (MacQueen, 1967) is a partitional clustering algorithm
- Each cluster has a cluster center, called centroid
- K is specified by the user

K-means Algorithm

- Given k:
  - Choose *k* (random) data points (seeds) to be the initial centroids, cluster centers
  - Assign each data point to the closest centroid
  - Re-compute the centroids using the current cluster memberships
  - If a convergence criterion is not met, repeat steps 2 and 3

Measure the Distance

- Similarity measure (distance measure)
  - Euclidean distance $d(x, y) = \sqrt{(x - y)^2} = \sqrt{\sum_{i=1}^{d}(x_i - y_i)^2}$
  - Manhattan distance $d(x, y) = |x - y| = \sum_{i=1}^{d}|x_i - y_i|$

Stopping Criterion

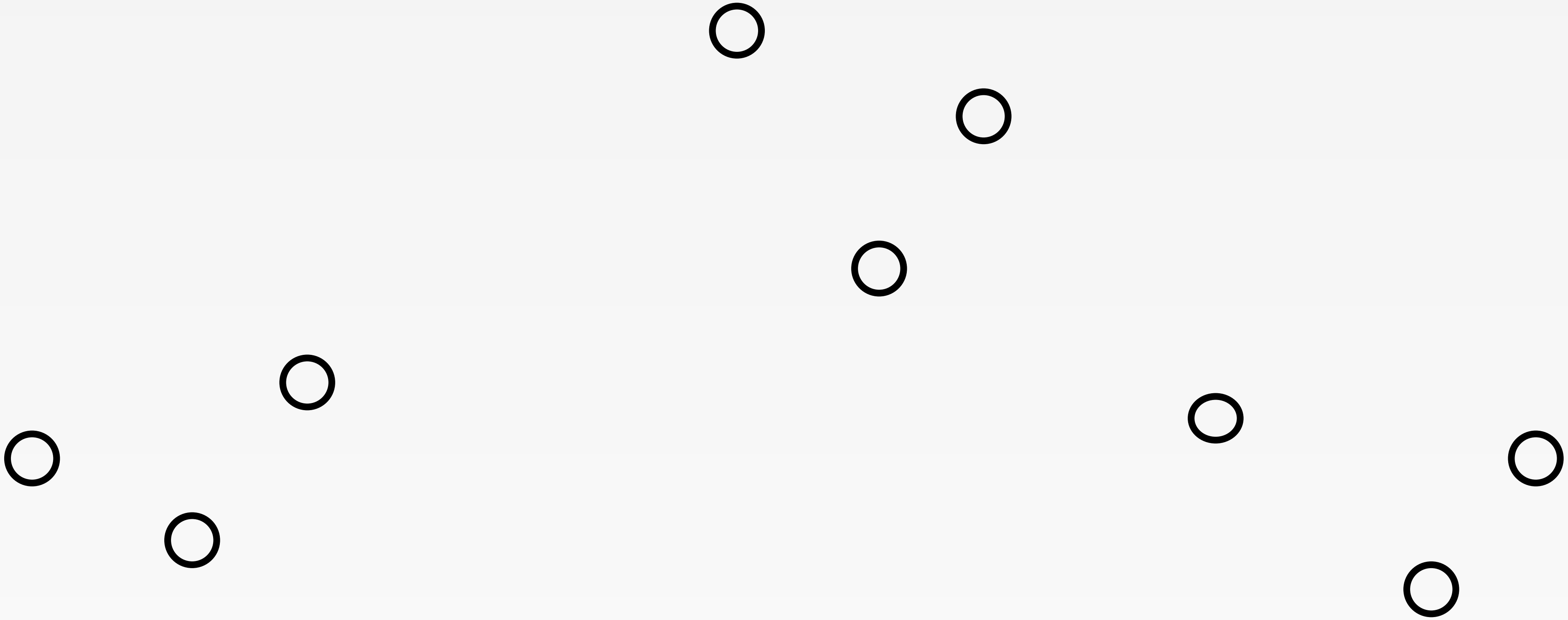
- no (or minimum) re-assignments of data points to different clusters, *or*
- no (or minimum) change of centroids, or
- minimum decrease in the **sum of squared error**(SSE),

$$SSE = \sum_{j=1}^{k} \sum_{x \in C_j} d(x, m_j)^2$$

- C_j is the jth cluster,
- m_j is the centroid of cluster C_j (the mean vector of all the data points in C_j)
- $d(x, m_j)$ is the distance between data point x and centroid m_j

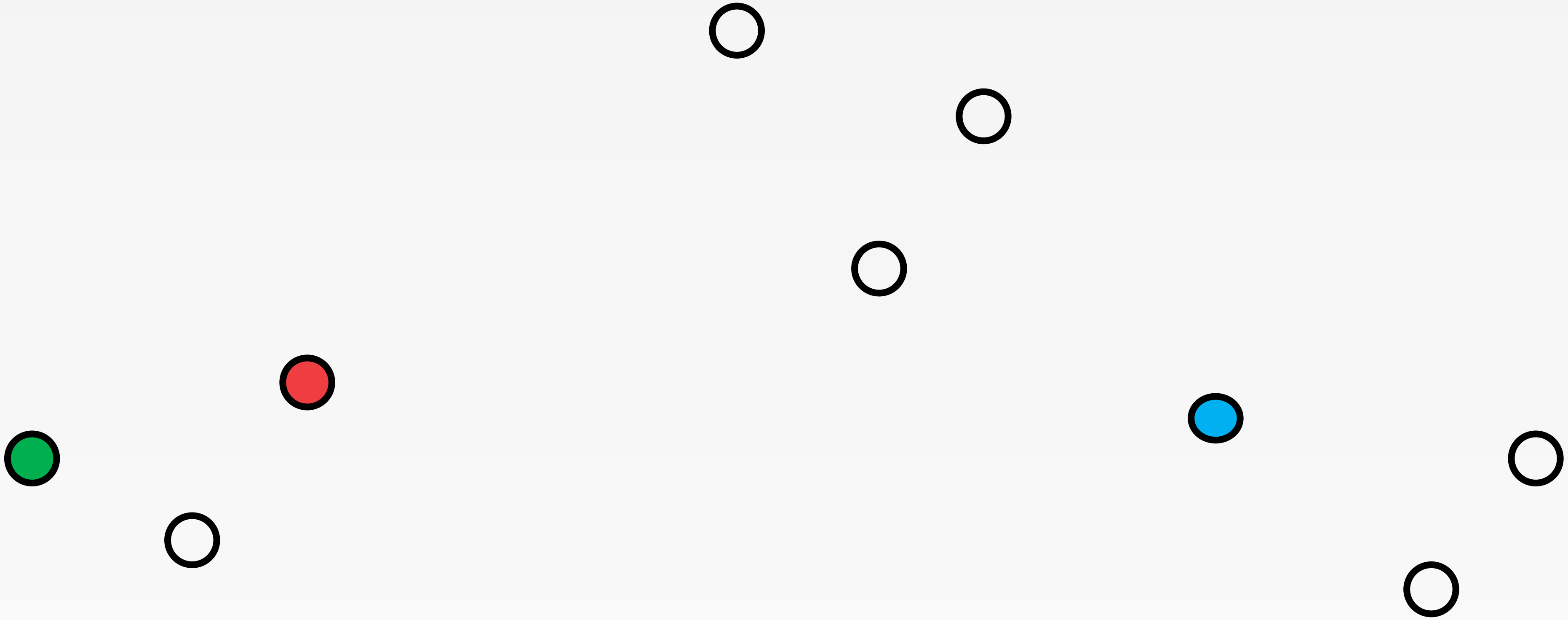
Illustrative Example

Given a set of data points



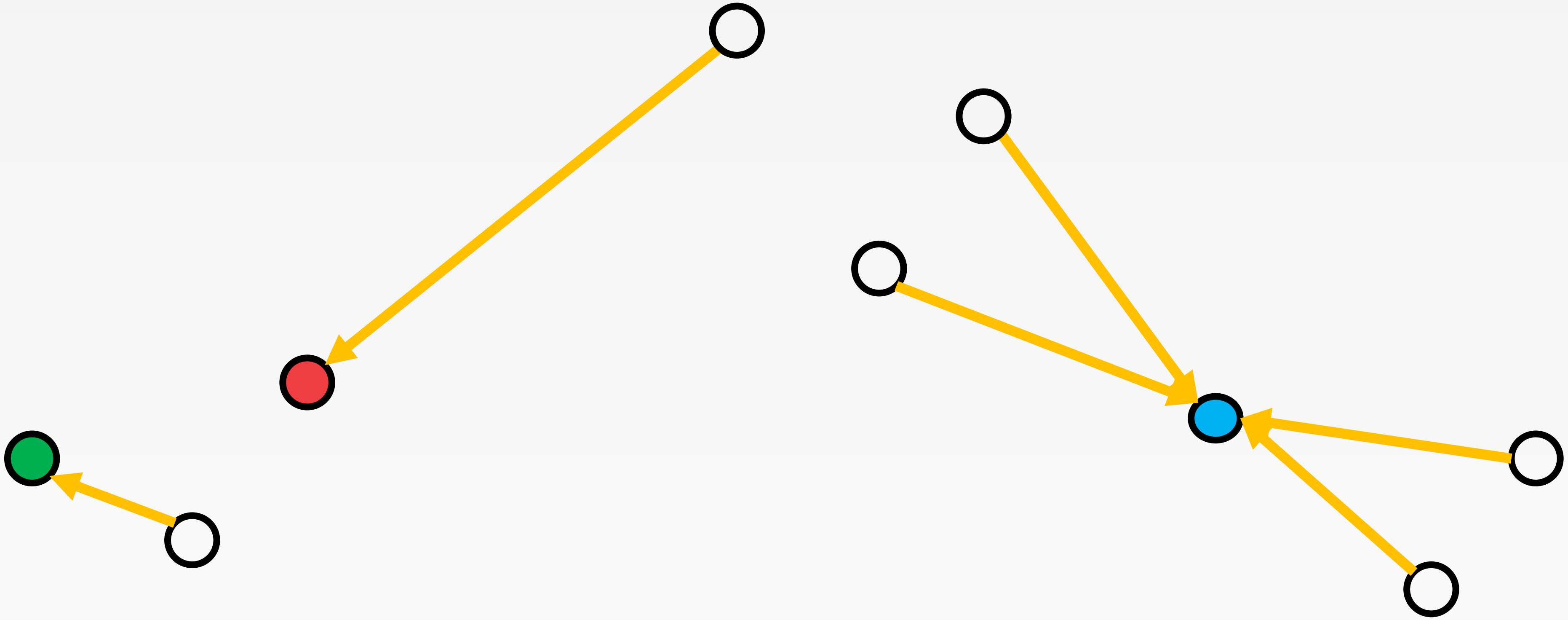
Illustrative Example

Select initial centers at random (k=3)



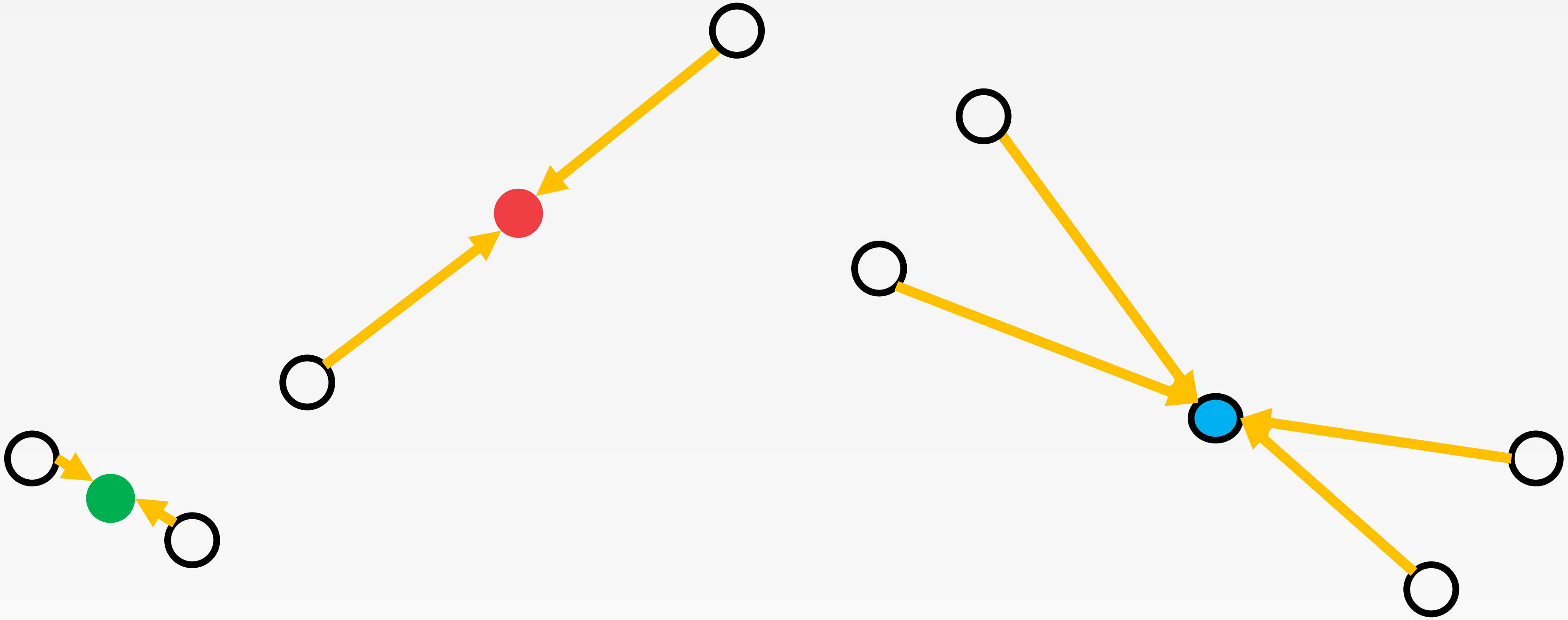
Illustrative Example

Assign each point to its nearest center



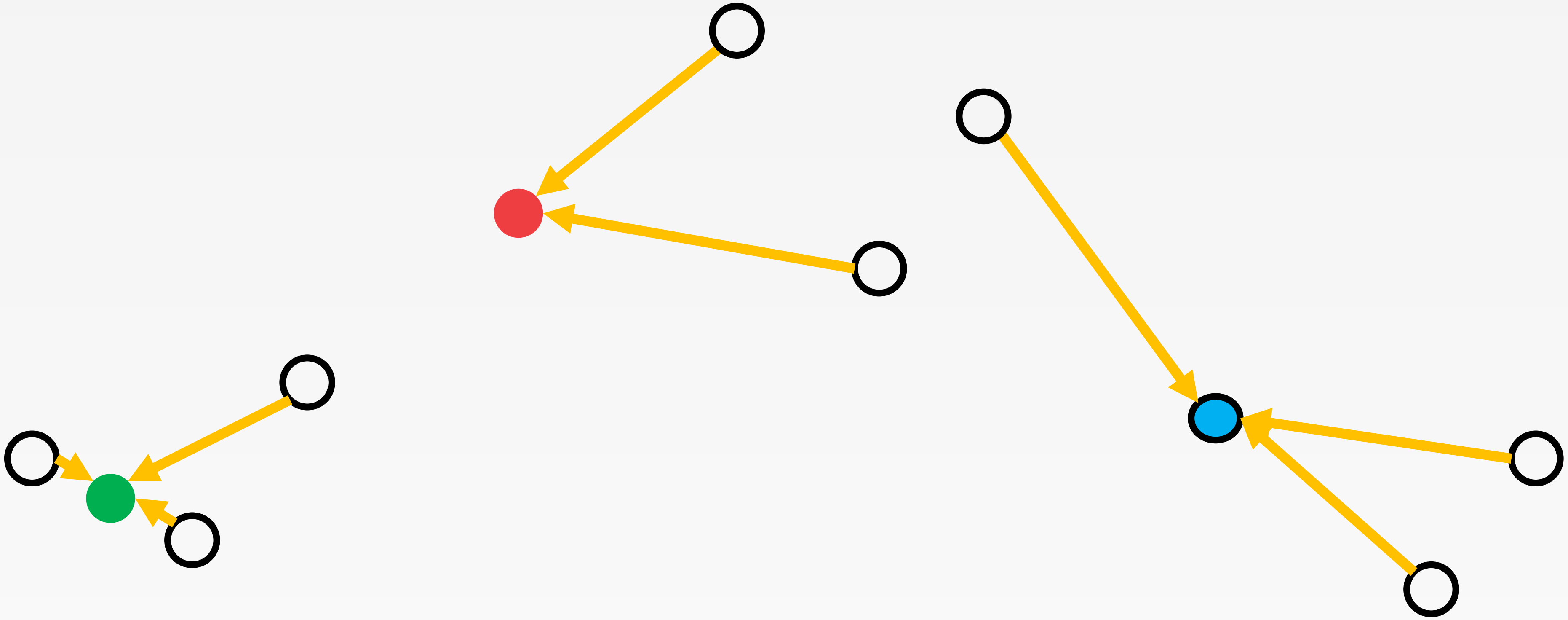
Illustrative Example

Recompute optimal centers given a fixed clustering



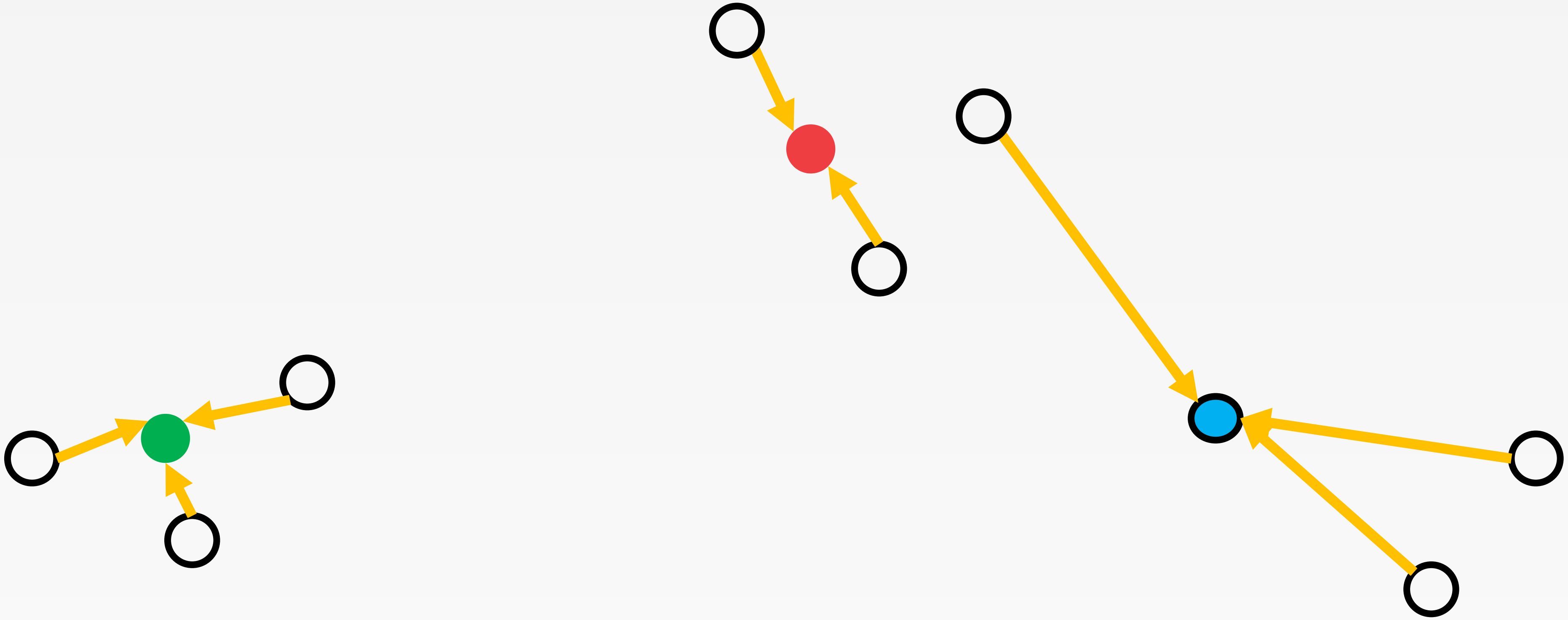
Illustrative Example

Assign each point to its nearest center



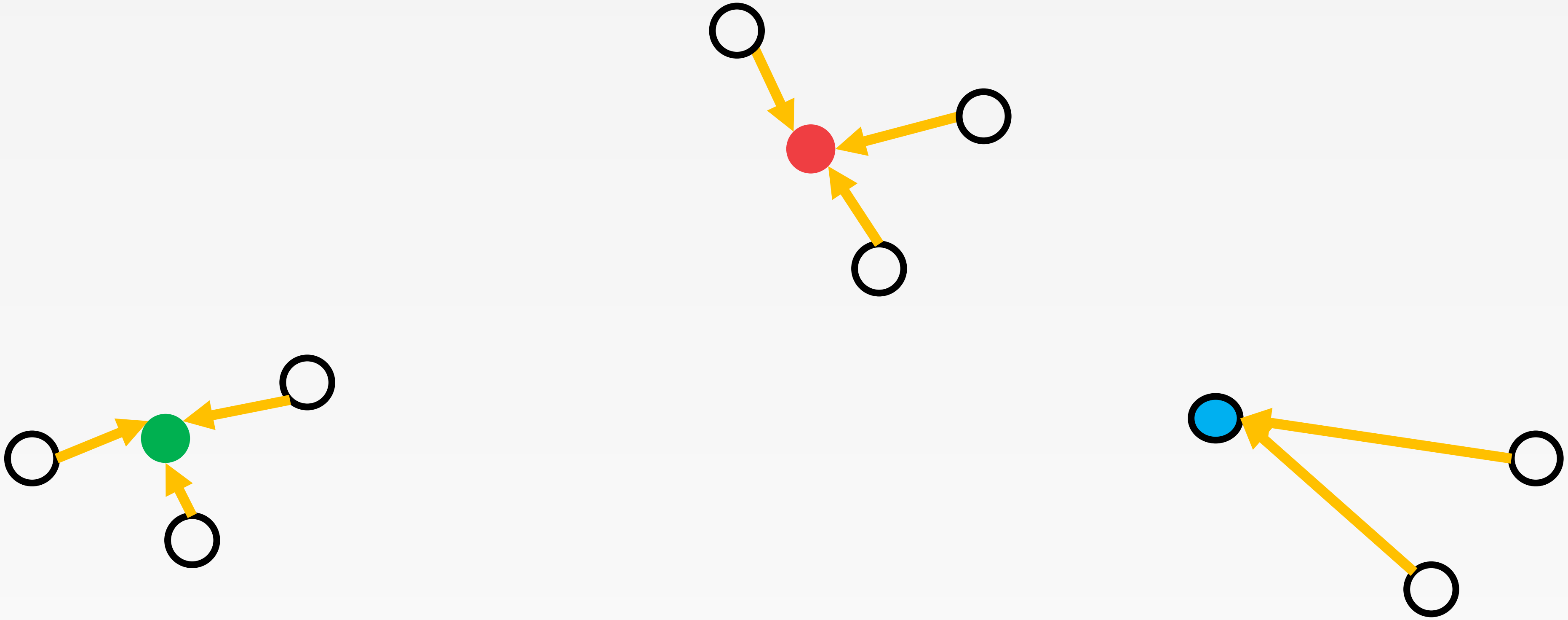
Illustrative Example

Recompute optimal centers given a fixed clustering



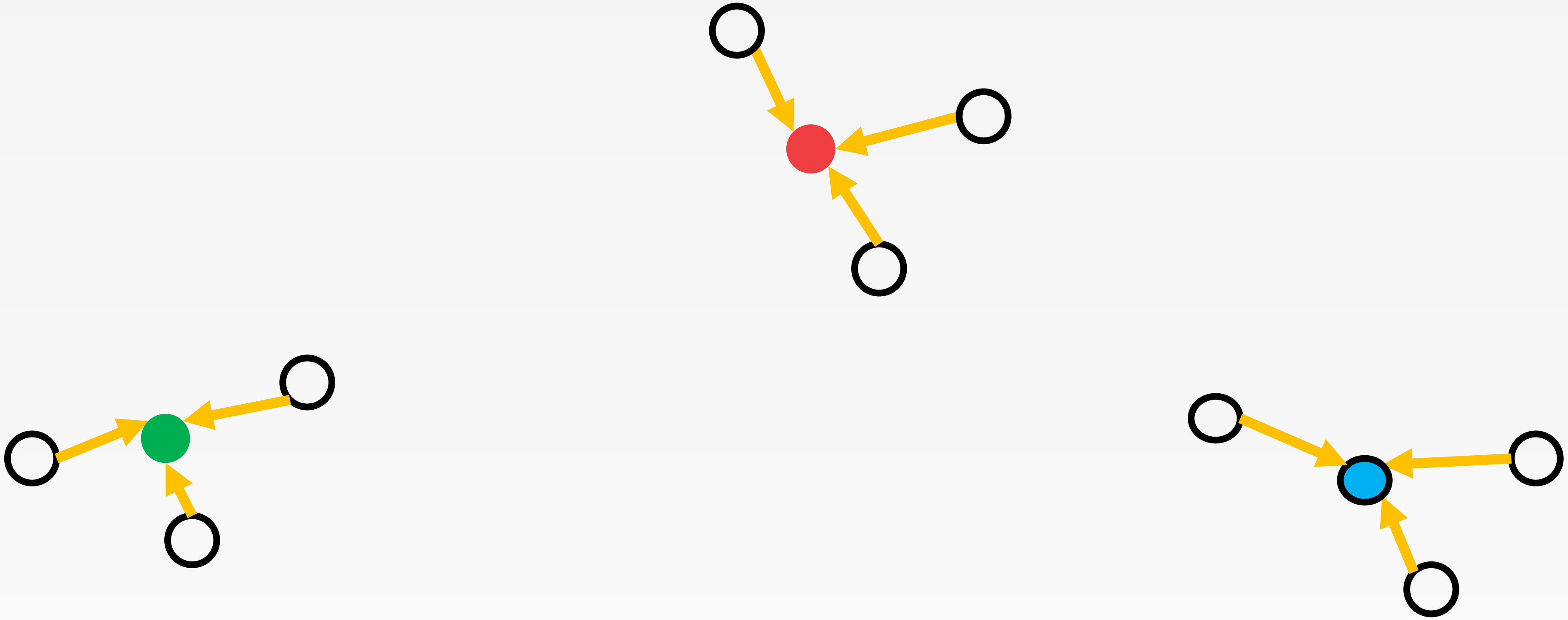
Illustrative Example

Assign each point to its nearest center



Illustrative Example

Recompute optimal centers given a fixed clustering



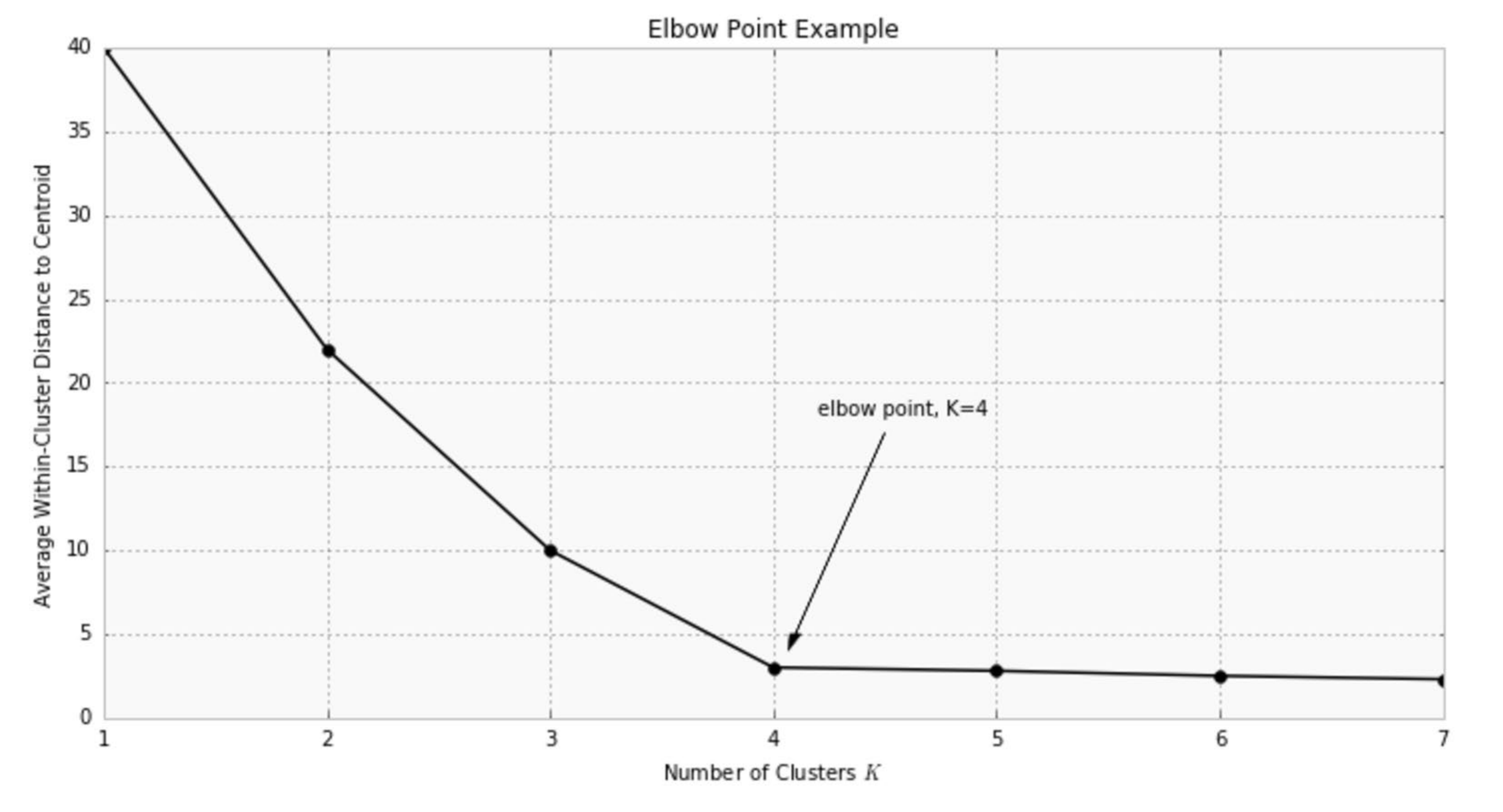
How to choose k?

Elbow method:

run k-means clustering on the dataset for a range of values of *k*

for each value of *k* calculate the sum of squared errors (SSE)

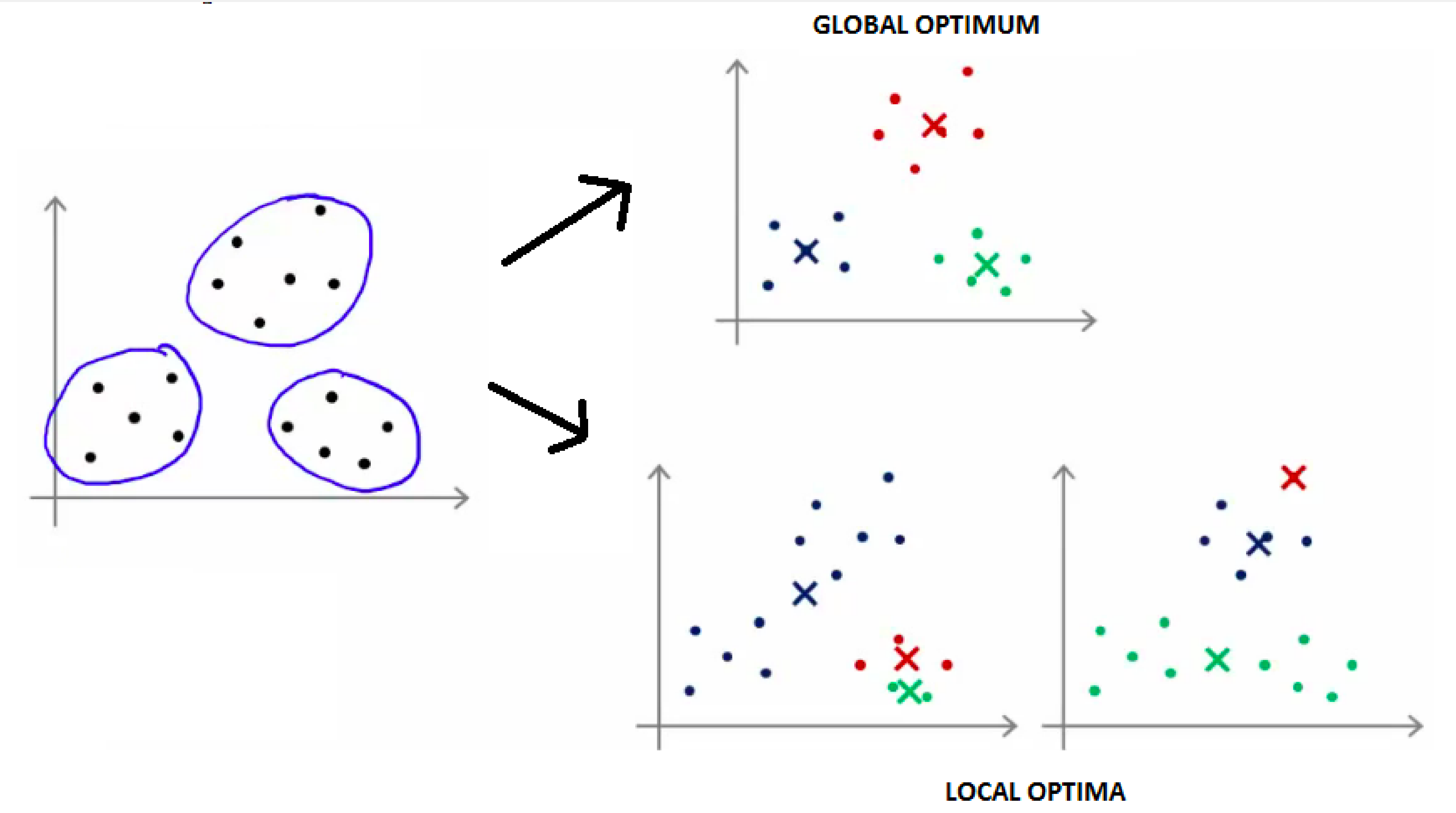
If the line chart looks like an arm, then the "elbow" on the arm is the value of *k* that is the best

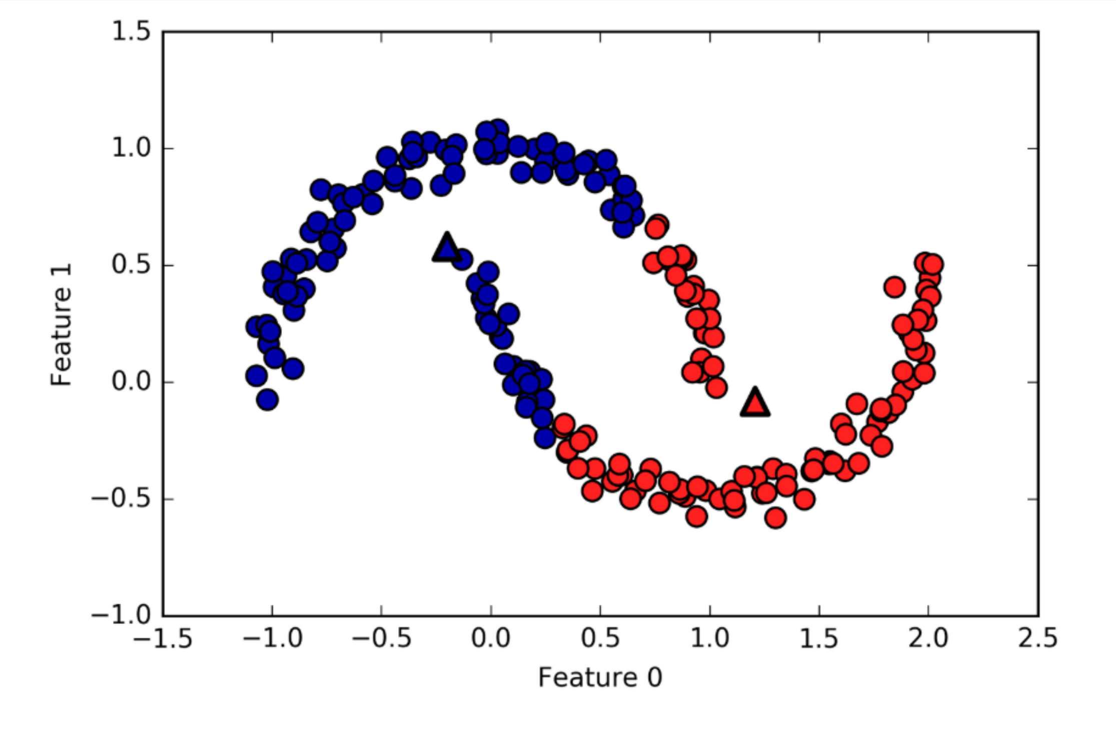


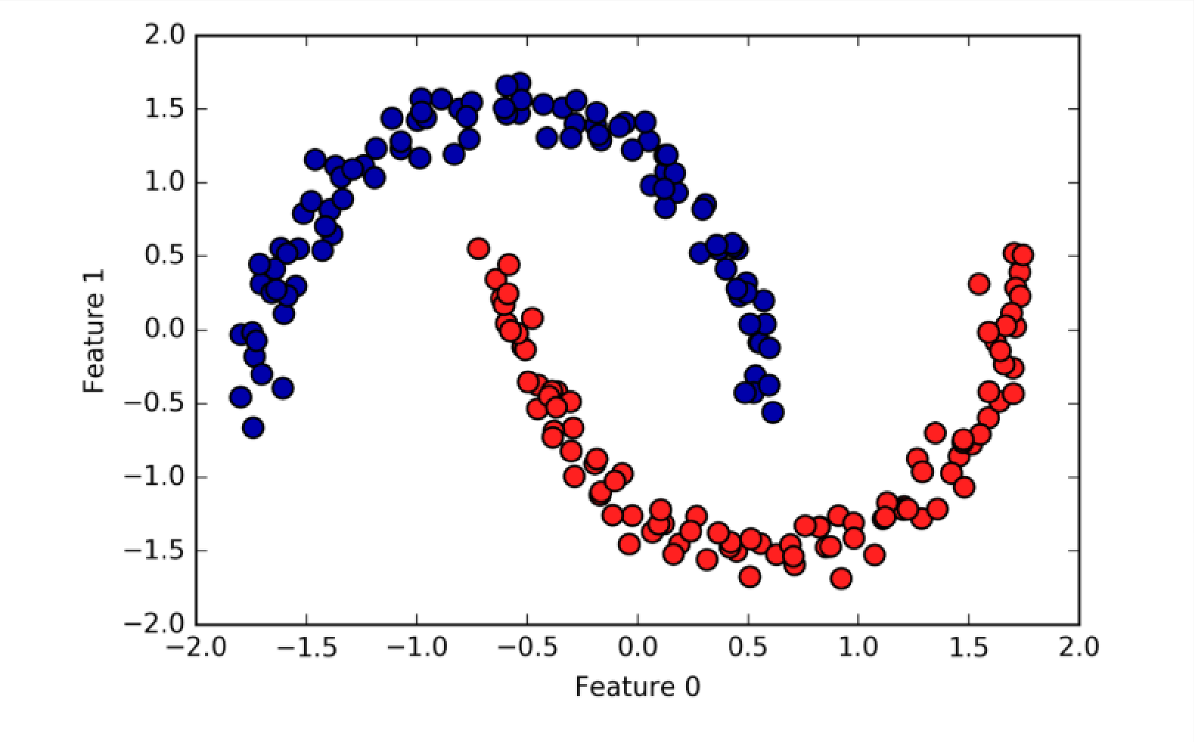
Elbow Point Example
Average Within-Cluster Distance to Centroid
40
35
30
25
20
15
10
5
0
elbow point, K=4
1
2
3
4
5
6
7
Number of Clusters K

Pros and Cons

- Strengths:
  - Simple: each to understand and to implement
  - Efficient
- Weakness:
  - The algorithm is sensitive to outliers
  - it terminates at a local optimum if SSE is used. The global optimum is hard to find due to complexity
    - Might be sensitive to initial seeds
  - Only simple cluster shapes


GLOBAL OPTIMUM
LOCAL OPTIMA


1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
Feature 0
Feature 1


2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
Feature 0
Feature 1

DBSCAN

Density-Based Spatial Clustering of Applications with Noise

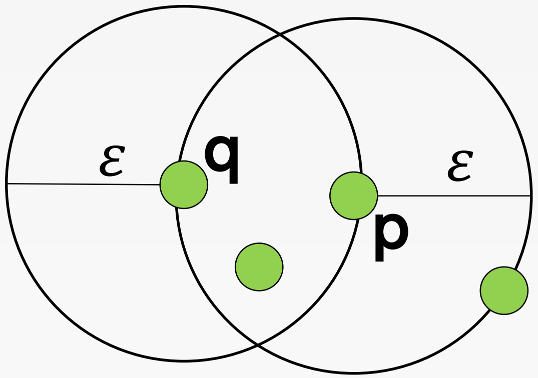
Source: https://cse.buffalo.edu/~jing/cse601/fa12/materials/clustering_density.pdf

Density-based Clustering

- Basic Idea:
  - Clusters are dense regions in the data space, separated by regions of lower object density
  - A cluster is defined as a maximal set of density-connected points

Density Definition

- ε-Neighborhood – Objects within a radius of $\boldsymbol{\varepsilon}$ from an object
$$N_\varepsilon(p): \{q \mid d(p,d) \le \varepsilon\}$$
- "High density" -- ε-Neighborhood of an object contains at least $\boldsymbol{MinPts}$ of objects

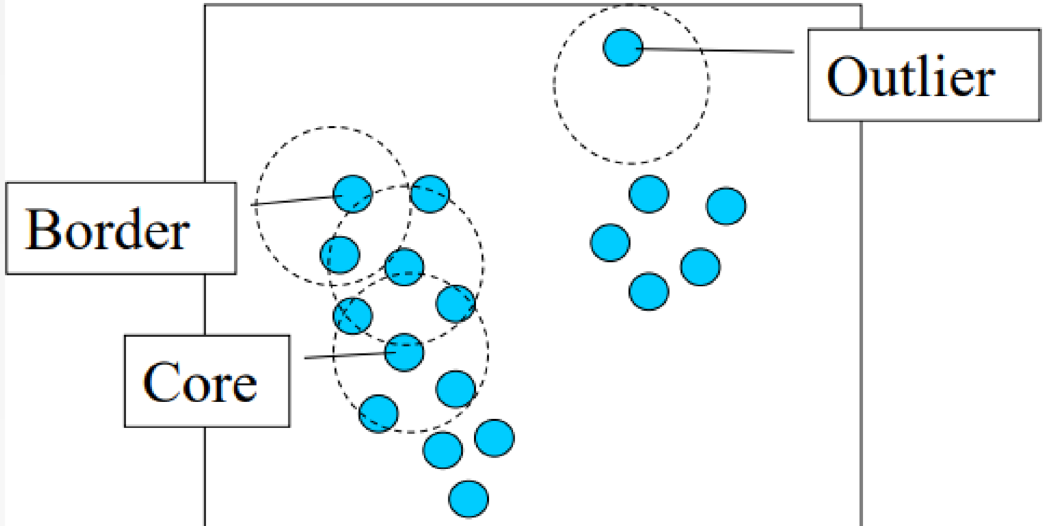


Density of p is "high" ($MinPts$ = 4)

Density of q is "low" ($MinPts$ = 3)

Core, Border, Outlier

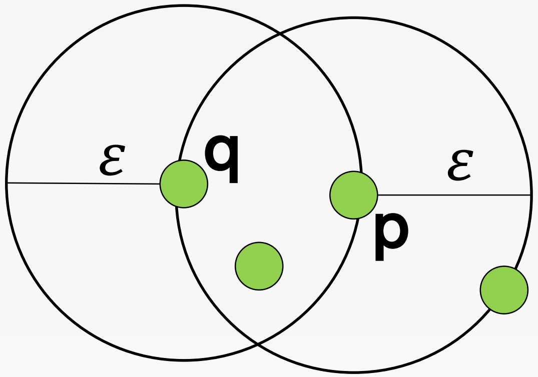
- Given ε and $MinPts$, categorize the objects into three exclusive groups:
  - Core point: has more than $MinPts$ points within ε (these are points that are at the interior of a cluster)
  - Border point: has fewer than $MinPts$ within ε, but is the neighborhood of a core point
  - Noise point: any point that is neither a core nor a border point



ε = 1unit, MinPts = 5

Density-reachability

- An object q is directly density-reachable from object p if p is a core object and q is in p's ε-neighborhood.



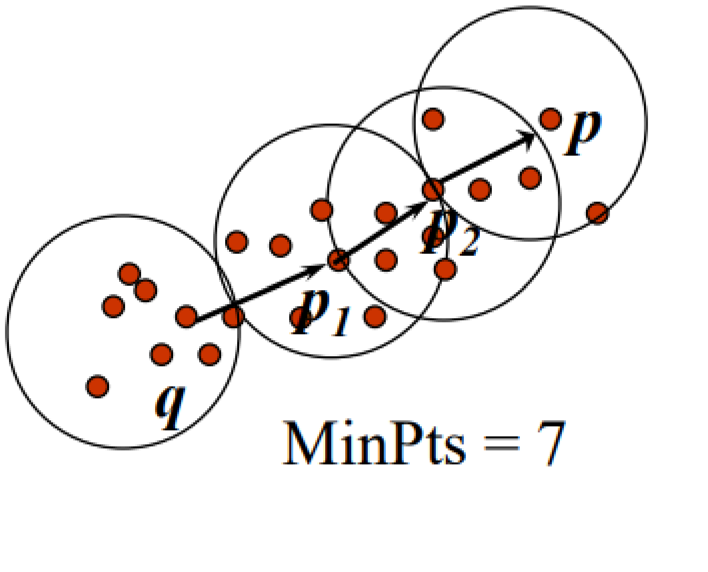
$MinPts$=4

q is directly density-reachable from q

p is not directly density-reachable from q

Density-reachability is asymmetric

Density-reachability



A point p is directly density-reachable from p_2

p_2 is directly density-reachable from p_1

p_1 is directly density-reachable from q

$p \leftarrow p_2 \leftarrow p_1 \leftarrow q$ form a chain

DBSCAN Algorithm

for each $o \in D$ **do**

    if o is not yet classified **then**

        if $|o$**'s** ε-neighborhood$|$ < $MinPts$

            assign o to *NOISE*

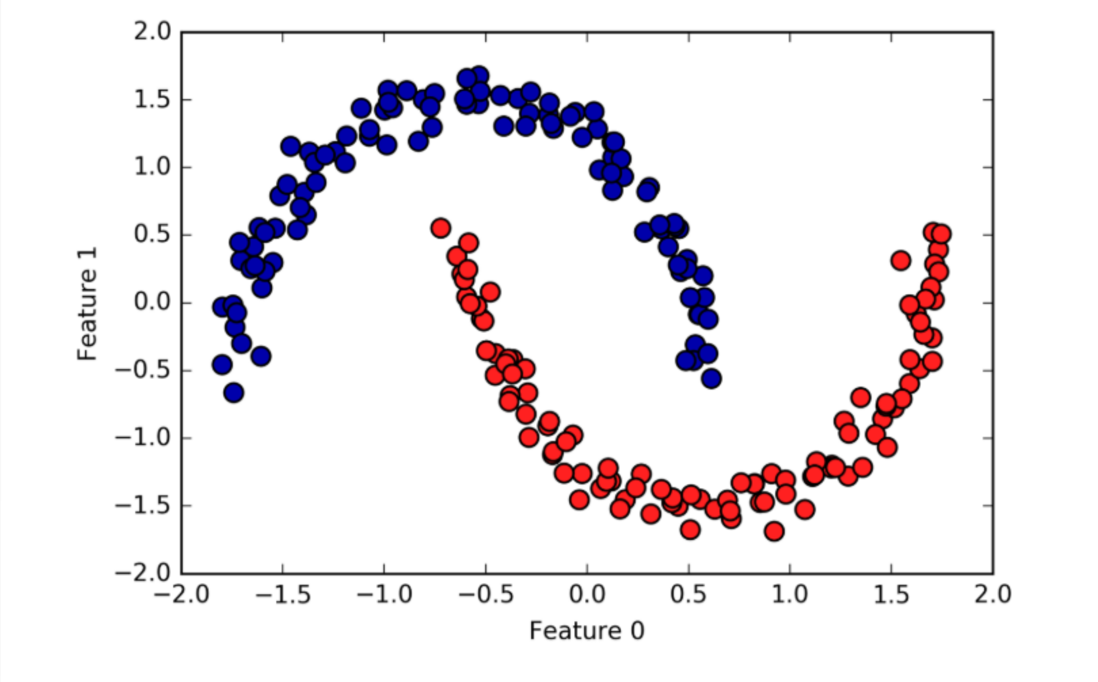
        else

            collect all objects density-reachable from o

            and assign them to a new cluster

Pros and Cons

- Can learn arbitrary cluster shapes (resistant to noise)
- Can detect outliers
- Needs two parameters to adjust



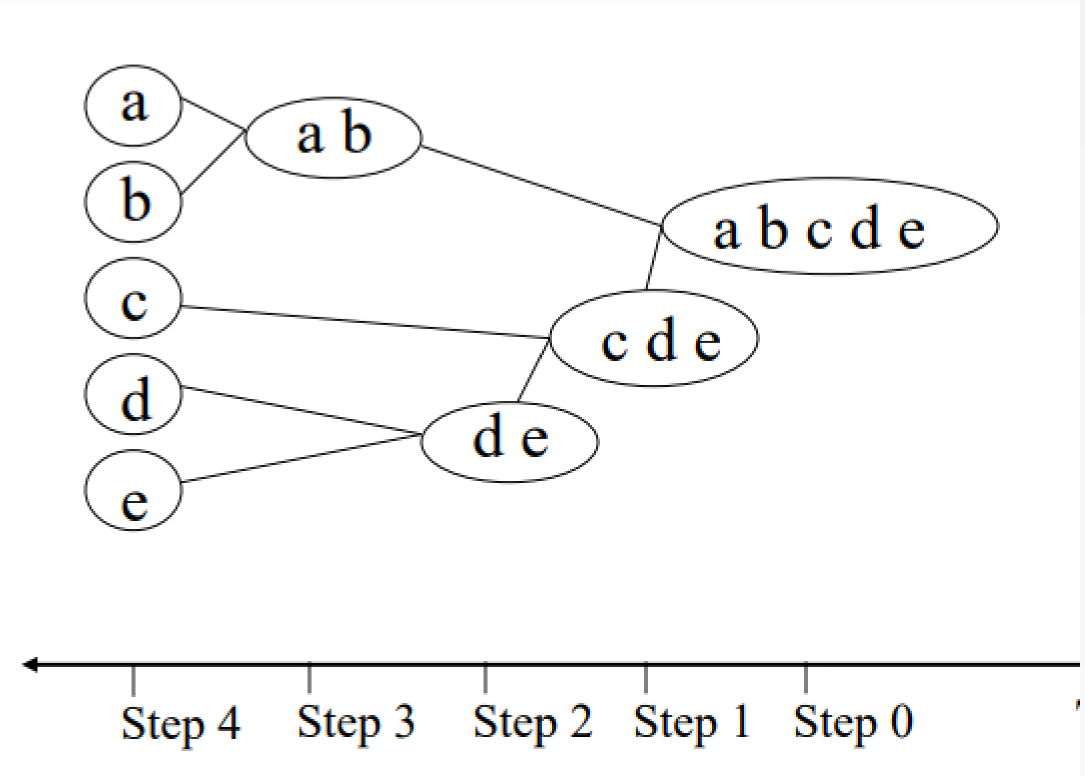
UT DALLAS
Naveen Jindal
School of Management

Hierarchical Clustering

https://cse.buffalo.edu/~jing/cse601/fa12/materials/clustering_hierarchical.pdf

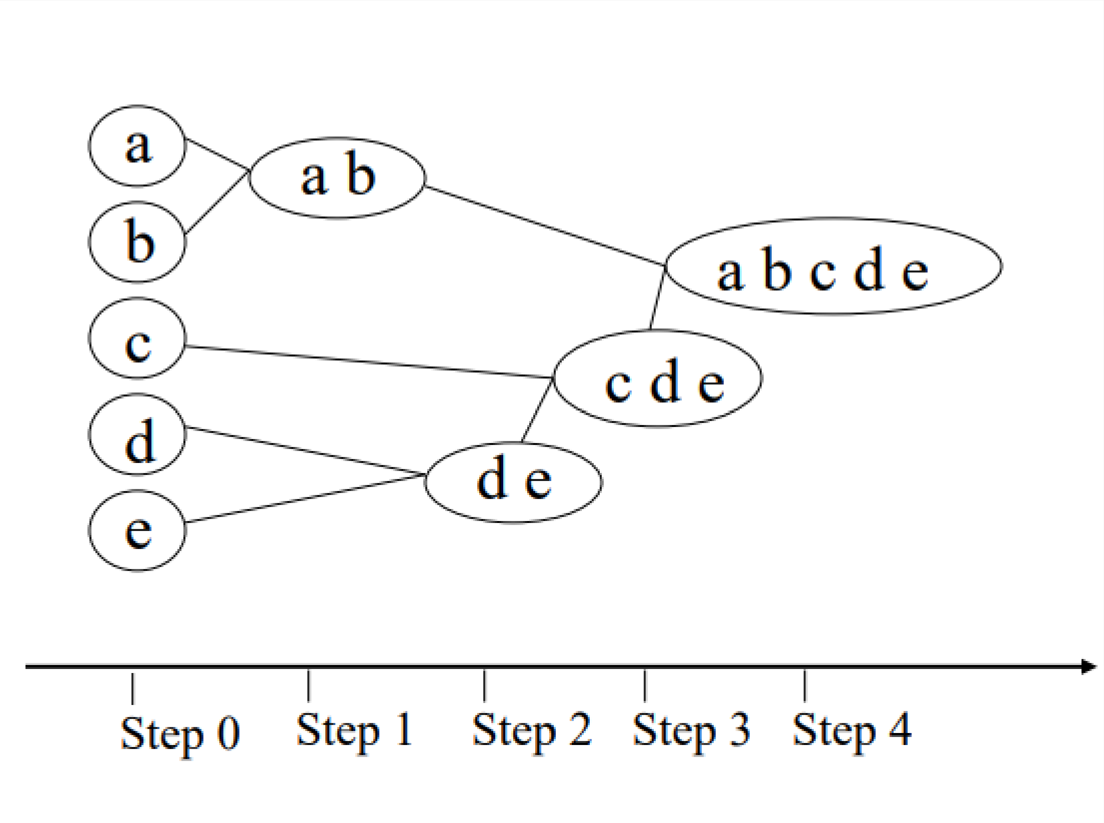
Types

- Divisive (top-down) clustering
  - All objects in one cluster
  - Select a cluster and split it into two sub clusters
  - Until each leaf cluster contains only one object



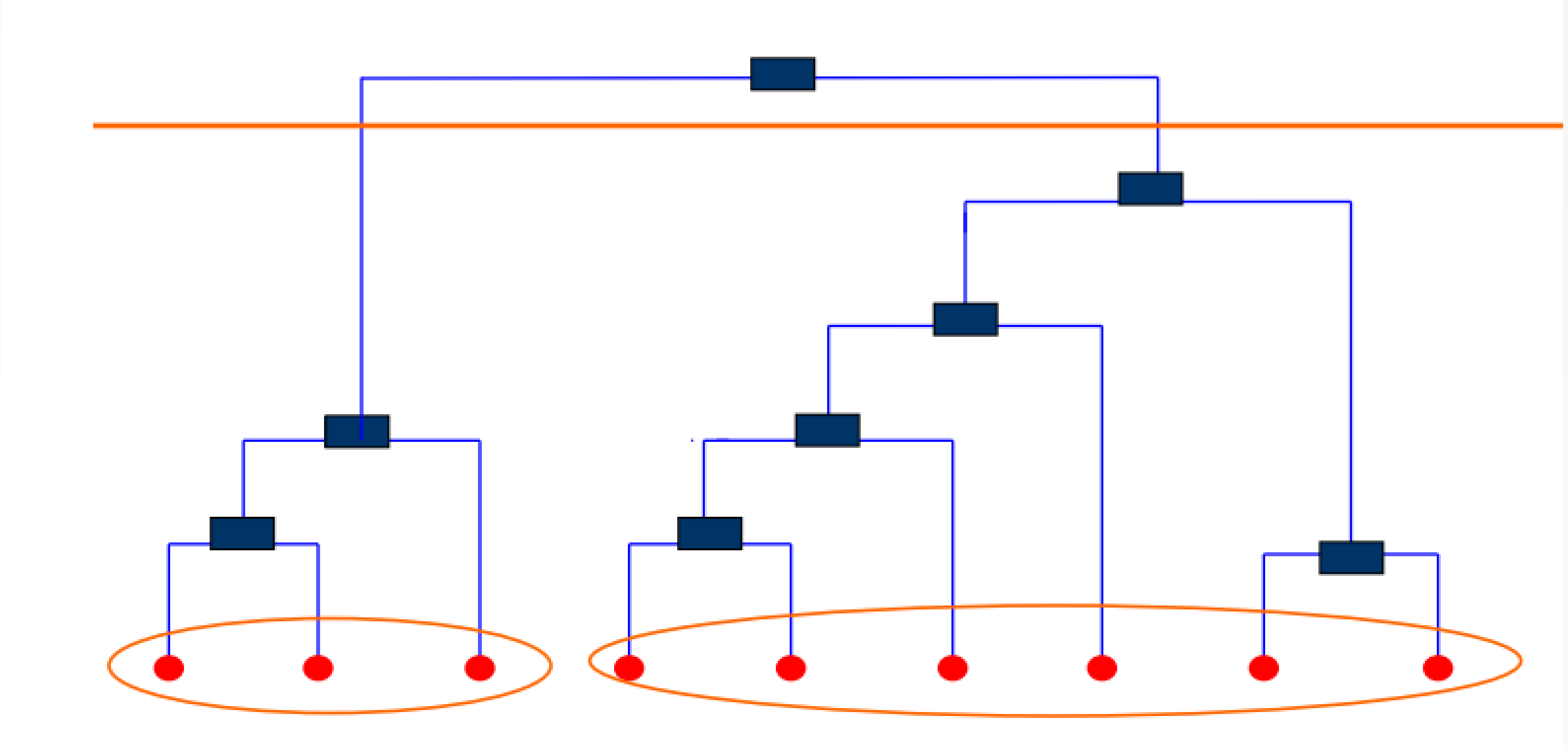
Types

- Agglomerative (bottom-up) clustering
  - Each object is a cluster
  - Merge two clusters which are most similar to each other
  - Until all objects are merged into a single cluster

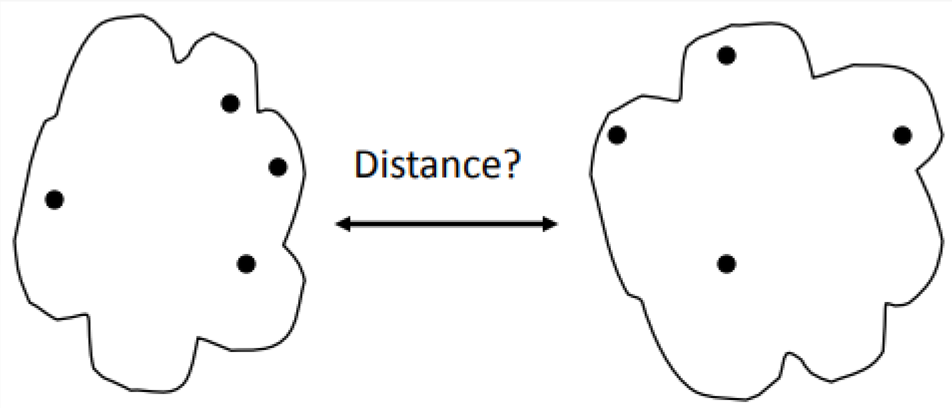


Dendrogram

- A tree that shows how clusters are merged/split hierarchically
- Each node on the tree is a cluster; each leaf node is a singleton cluster
- A clustering of the data objects is obtained by cutting the dendrogram at the desired level, then each connected component forms a cluster

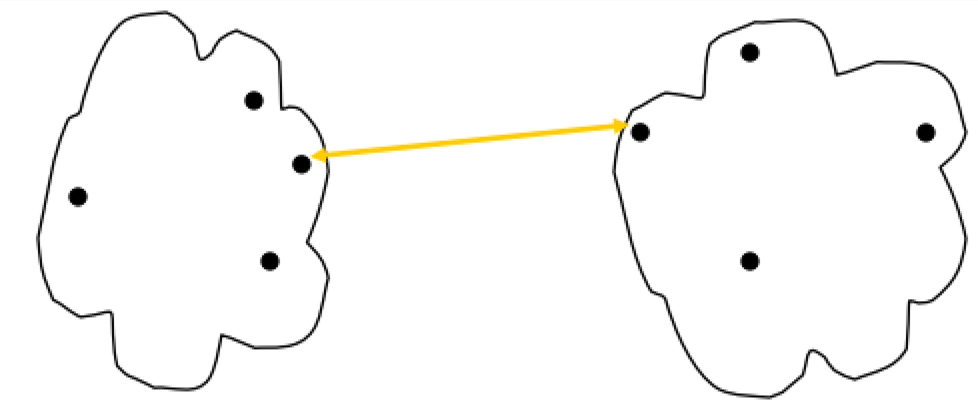


Inter-Cluster Distance

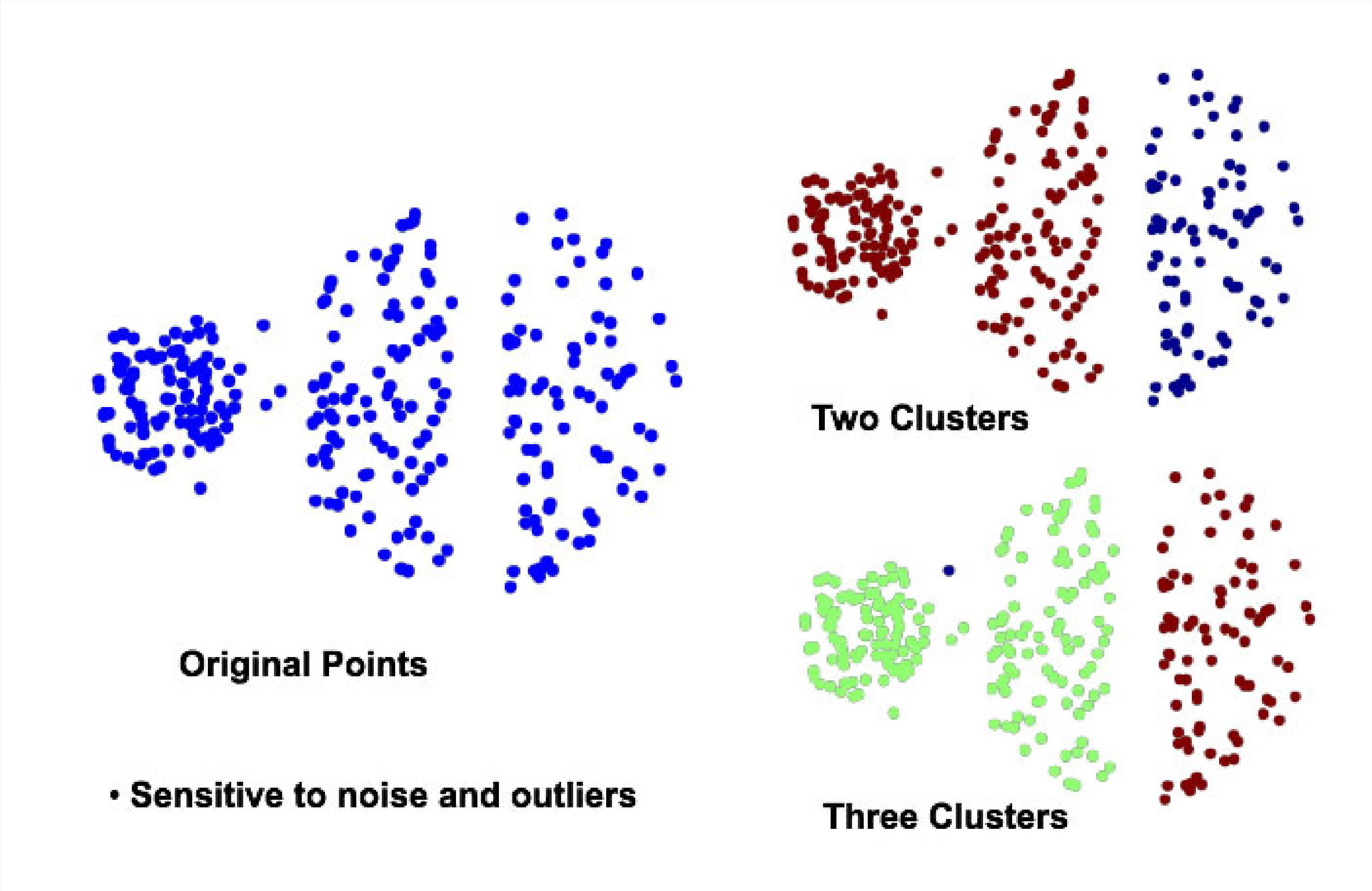


MIN (Single Link)

- The distance between two clusters is represented by the distance of the closest pair of data objects belonging to different clusters.
- Determined by one pair of points, i.e., by one link in the proximity graph

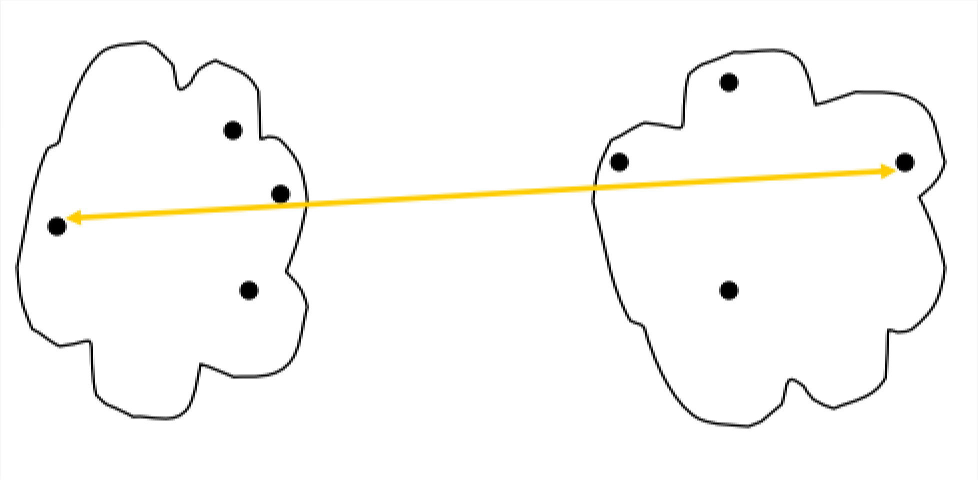


- Limitation: sensitive to noise/outliers


Original Points
• Sensitive to noise and outliers
Two Clusters
Three Clusters

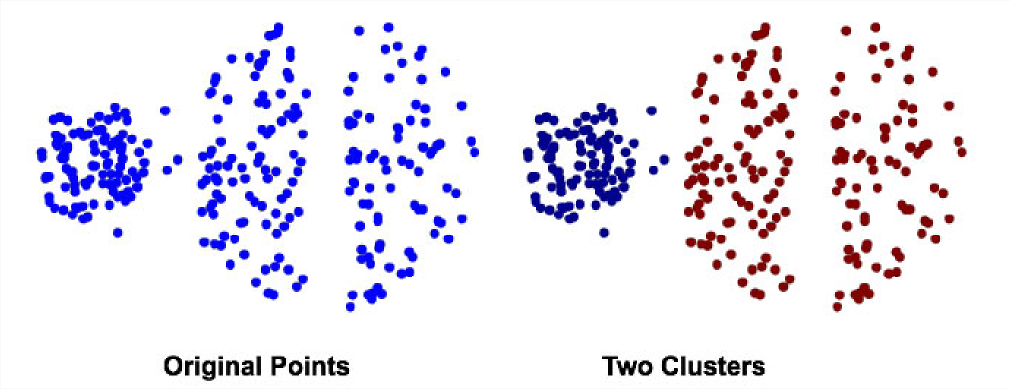
MAX (Complete link)

- The distance between two clusters is represented by the distance of the farthest pair of data objects belonging to different clusters

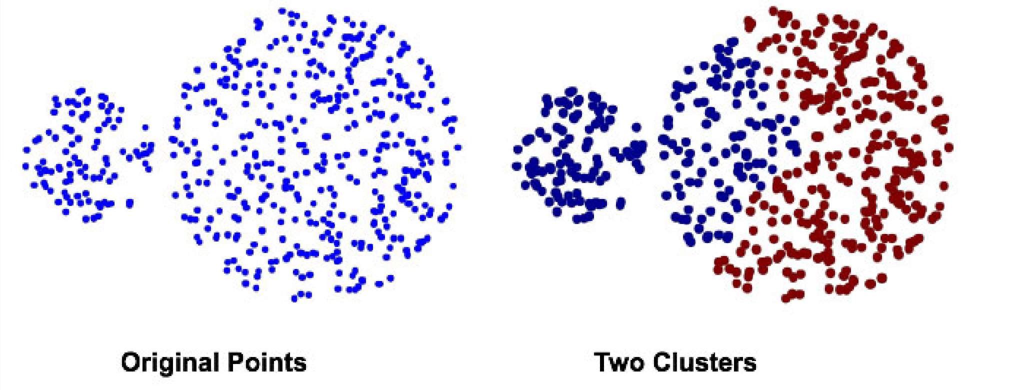


MAX (Complete link)

- Strength: less sensitive to noise/outliers

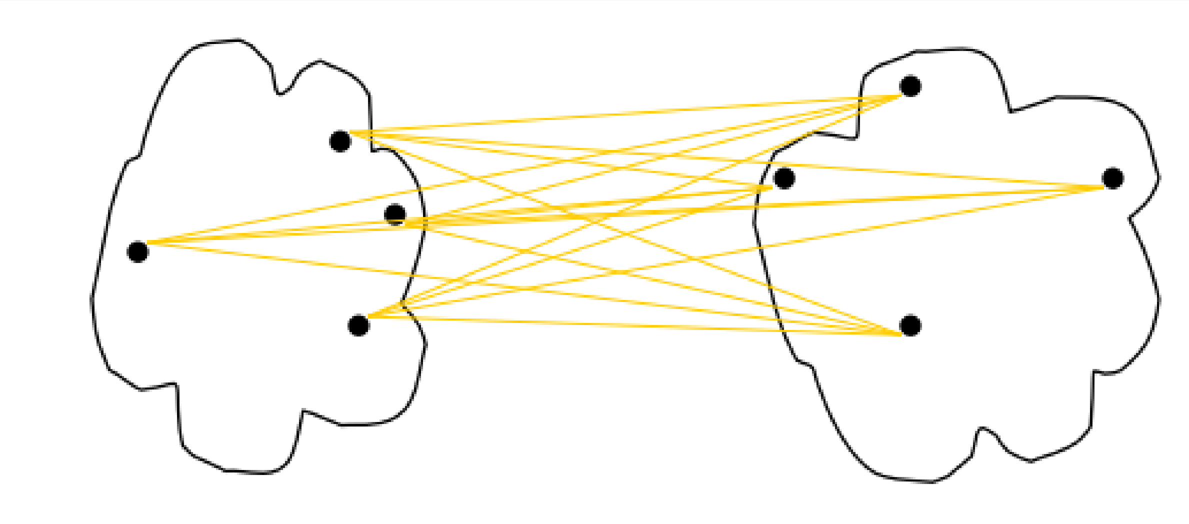


- Limitations: tends to break large clusters



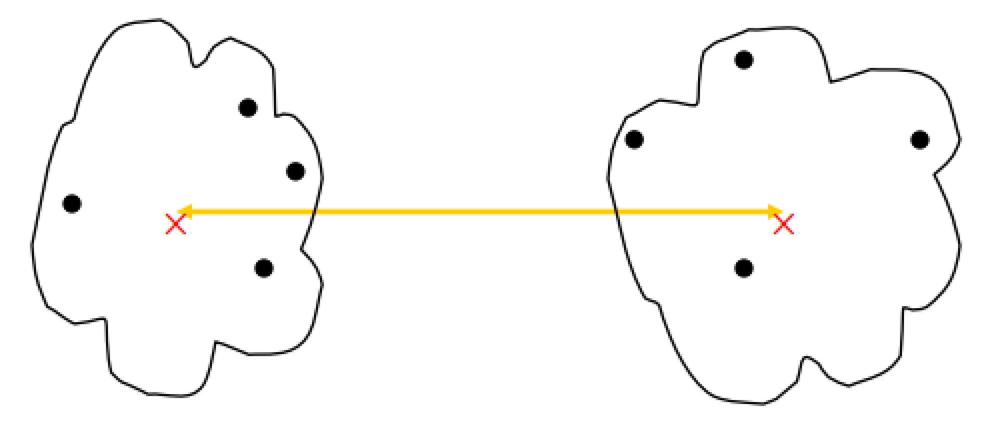
Group average

- The distance between two clusters is represented by the average distance of all pairs of data objects belonging to different clusters
- Determined by all pairs of points in the two clusters



Centroid Distance

- The distance between two clusters is represented by the distance between the centers of the clusters
- – Determined by cluster centroids



Ward's Method

- Similarity of two clusters is based on the increase in squared error when two clusters are merged
- Similar to group average if distance between points is distance squared
- Less susceptible to noise and outliers

Python Practice

Questions?

For Next Week…

- Principal component analysis