DATA MINING

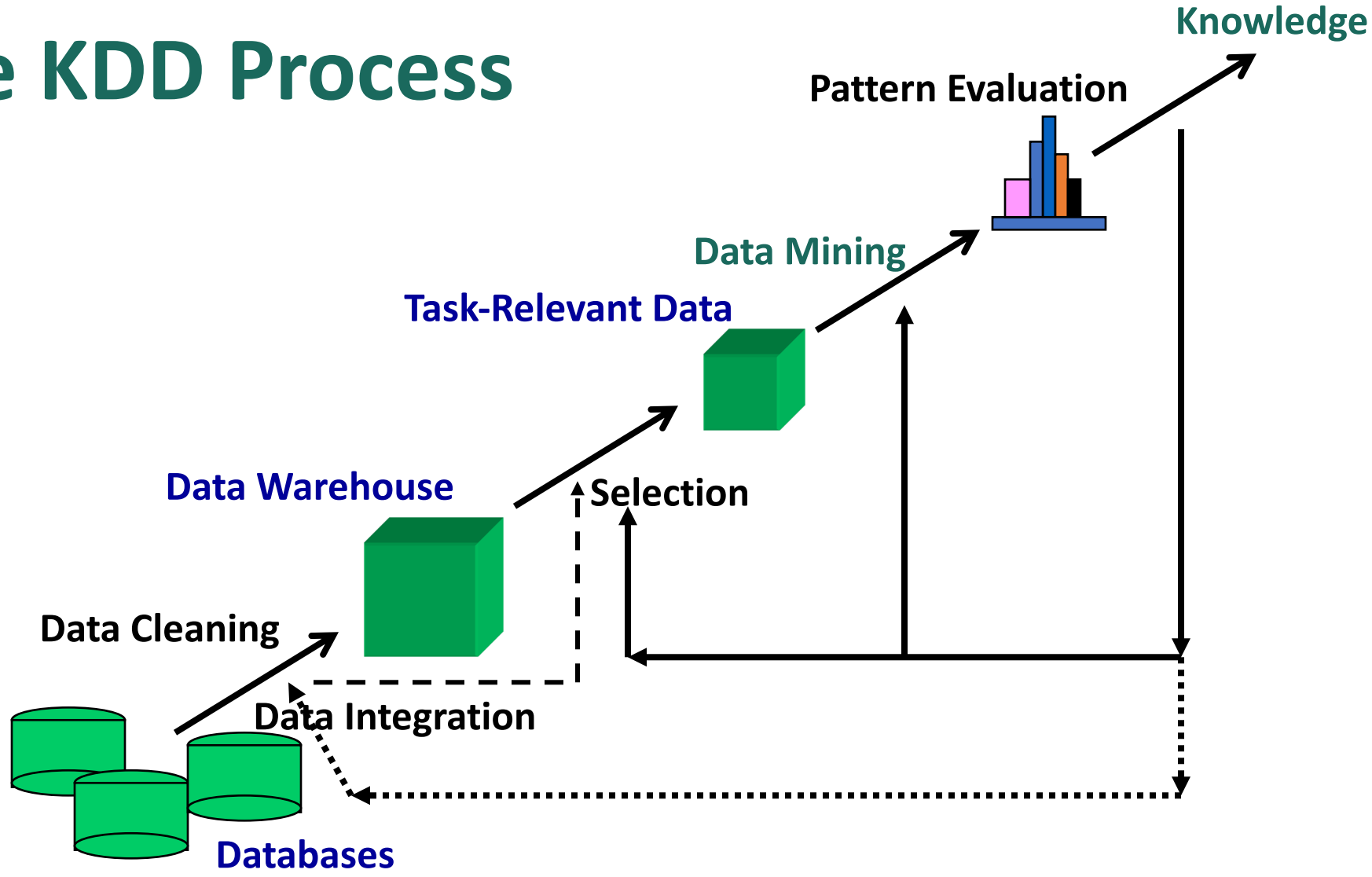
Clustering

Riccardo Guidotti, Anna Monreale, Salvatore Rinzivillo

What Is Data Mining?

- Data mining (knowledge discovery from data)
- Data mining is the use of efficient techniques for the analysis of very large collections of data and the extraction of useful and possibly unexpected patterns in data (hidden knowledge).

The KDD Process



Data Mining Tasks...

- Clustering
- Classification
- Pattern Mining

Data

- Collection of data objects and their attributes
- An attribute is a property of an object
  - Examples: eye color of a person, temperature, etc.
  - Attribute is also known as variable, field, characteristic, or feature
- A collection of attributes describe an object
  - Object is also known as record, point, case, sample, entity, or instance

Attributes

Objects

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Supervised vs. Unsupervised Learning

- **Supervised learning (classification)**
  - Supervision: The training data (observations, measurements, etc.) are accompanied by **labels** indicating the class of the observations
  - New data is classified based on the training set
- **Unsupervised learning (clustering)**
  - The class **labels** of training data is **unknown**
  - Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data

Clustering


KDD
Lab


IN SUPREMÆ DIGNITATIS
1343
UNIVERSITÀ DI PISA


SoBigData

Clustering Definition

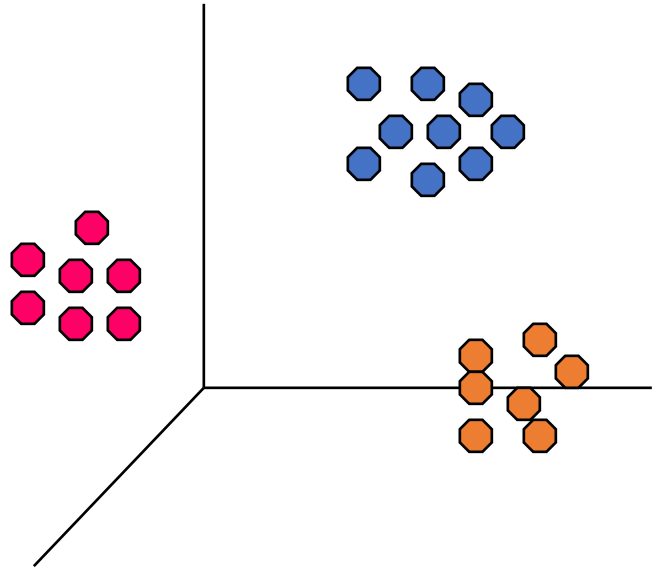
- **Cluster**: A collection of data objects
- Given a set of data points, each having a **set of attributes**, and a **similarity measure** among them, find clusters such that
  - Data points in one cluster are more similar to one another
  - Data points in separate clusters are less similar to one another
- **Similarity Measures?**
  - Euclidean Distance if attributes are continuous
  - Other Problem-specific Measure

Illustrating Clustering

Euclidean Distance Based Clustering in 3-D space.

Intracluster distances are minimized

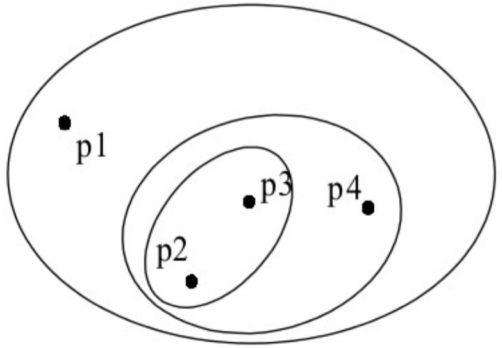
Intercluster distances are maximized



Different clustering approaches

- **PARTITIONING ALGORITHMS**
  Directly divides data points into some prespecified number of clusters without a hierarchical structure

- **HIERARCHICAL ALGORITHMS**
  Groups data with a sequence of nested partitions, either from singleton clusters to a cluster containing all elements, or viceversa

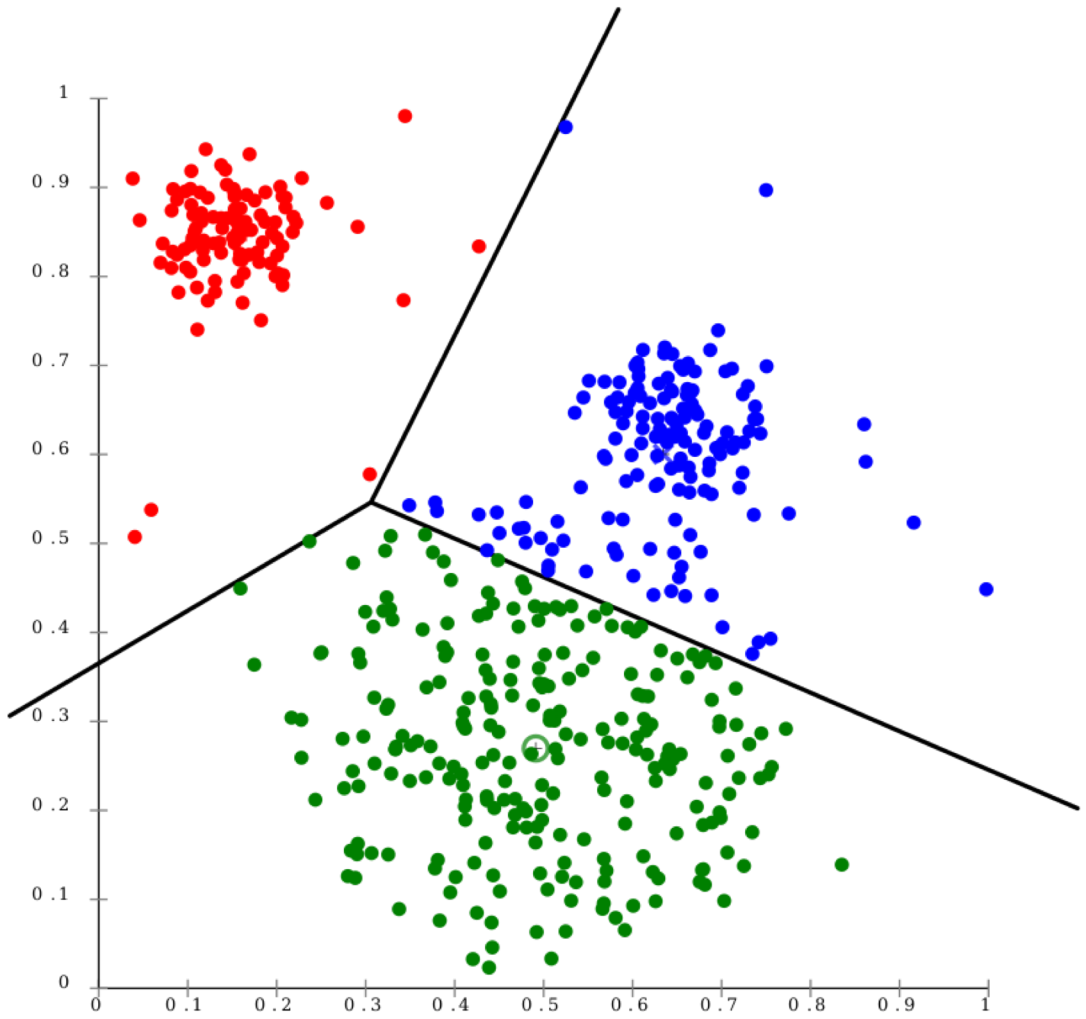


Center-based clustering

- A cluster is a set of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster

- The center of a cluster is often a centroid, the average of all the points in the cluster, or a medoid, the most “representative” point of a cluster

K-Means



Center-BASED Clustering: Application

- **Market Segmentation**

Goal: subdivide a market into distinct **subsets of customers** where any subset may conceivably be selected as a market target to be reached with a **distinct marketing mix.**

Approach

1. **Collect different attributes** of customers based on their geographical, **Demographic,** lifestyle, **Behavioral** related information
2. **Find clusters** of similar customers
3. **Measure the clustering quality** by observing buying patterns of customers in same cluster vs. those from different clusters.

K-Means Clustering

- Partitional clustering approach where each cluster is identified 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

K-Means Clustering

- Initial centroids are often chosen **randomly** → Clusters produced vary from one run to another
- The centroid is the **mean** of the points in the cluster
- **Closeness** is measured by a distance/similarity function: Euclidean distance, cosine similarity, correlation, etc.
- Typically convergence happens in the first few iterations
- Often the stopping condition is changed to: **Until relatively few points change clusters**

Clustering Evaluation

- **Sum of Squared Error (SSE)**

  - Measures the quality of the clustering computing for each point the the distance to the nearest cluster (**error**)

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

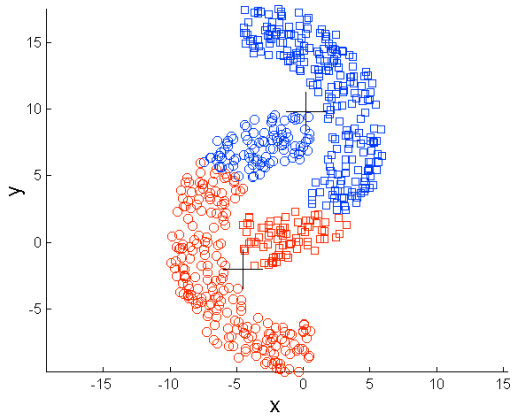
  - x is a data point in cluster C_i and m_i is centroid for cluster C_i

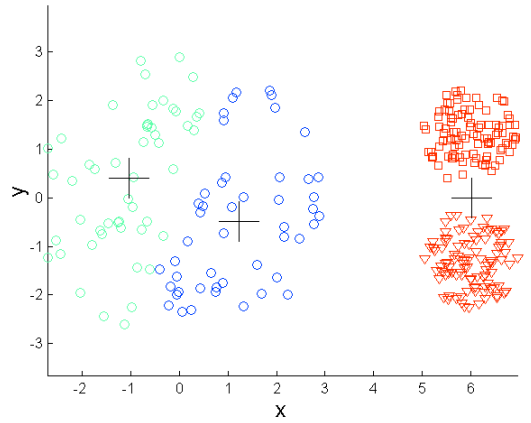
  - Smallest errors correspond to better clustering

  - SSE decreases with higher number of clusters (K)

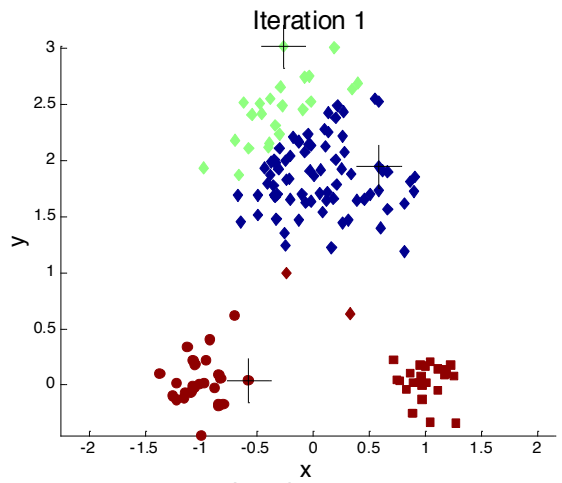
Limitations

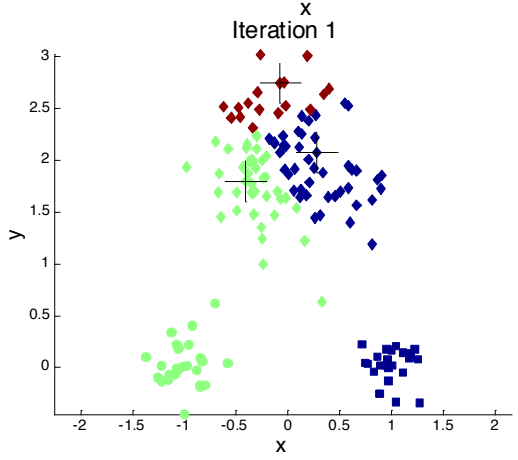
- K-means finds clusters with **globular shape** and has problems when data contain natural **non-globular groups**
- K-means has problems in finding clusters in data with **different densities**
- K-means is not able to capture **outliers** and noisy data

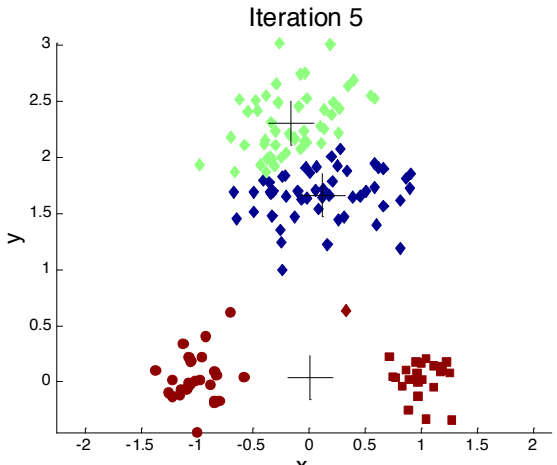


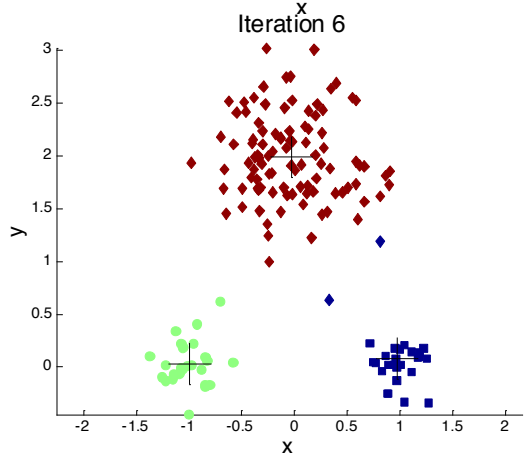


Initial centroids









- Different initial centroids may lead to different solutions
- Given K ‘real’ clusters, the chance of selecting one centroid from each cluster is very small

Initial centroids

- Different initial centroids may lead to different solutions
- Given K ‘real’ clusters, the chance of selecting one centroid from each cluster is very small

- **SOLUTIONS:**
  - Run k-means multiple times
  - Apply hierarchical clustering to determine initial centroids
  - Select more than k initial centroids and then select among these initial centroids

The problem of Empty Clusters

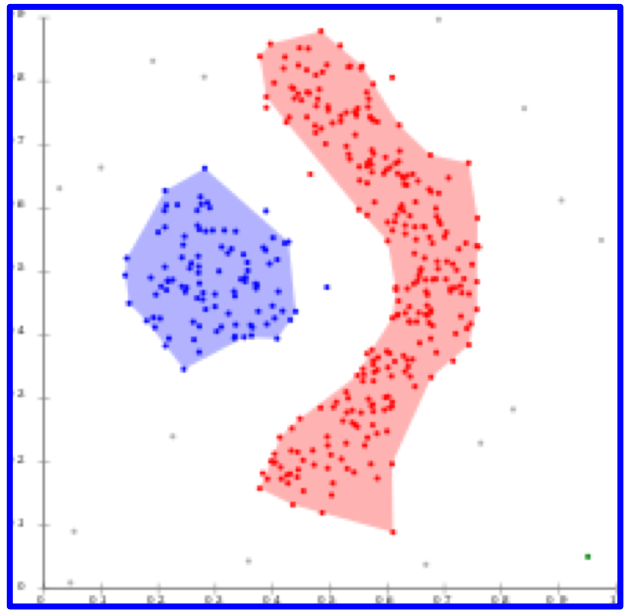
Basic K-means algorithm can yield empty clusters

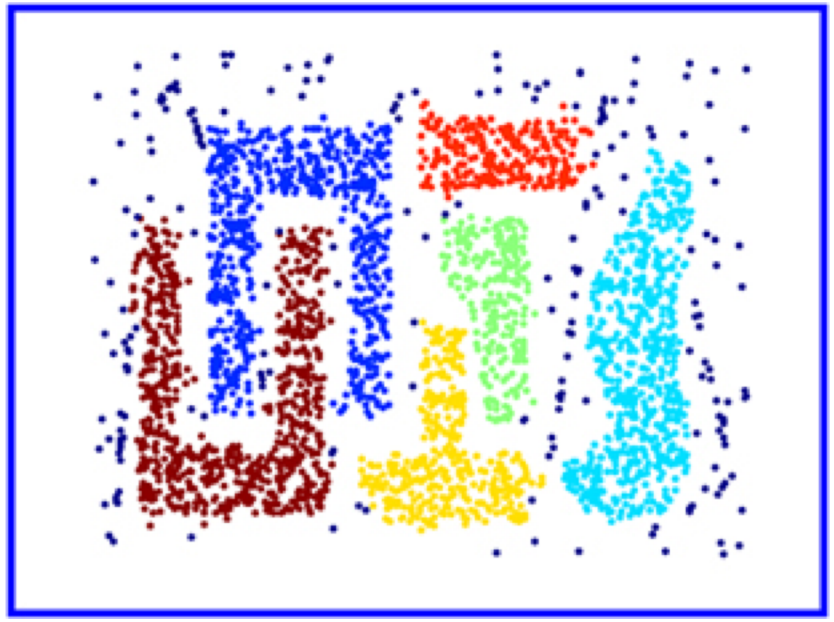
SOLUTIONS:

- Choose the point that contributes most to SSE
- Choose a point from the cluster with the highest SSE
- If there are several empty clusters, the above can be repeated several times.

Density-based clustering

Clusters are **dense regions** in the data space seprated by regions with lower density

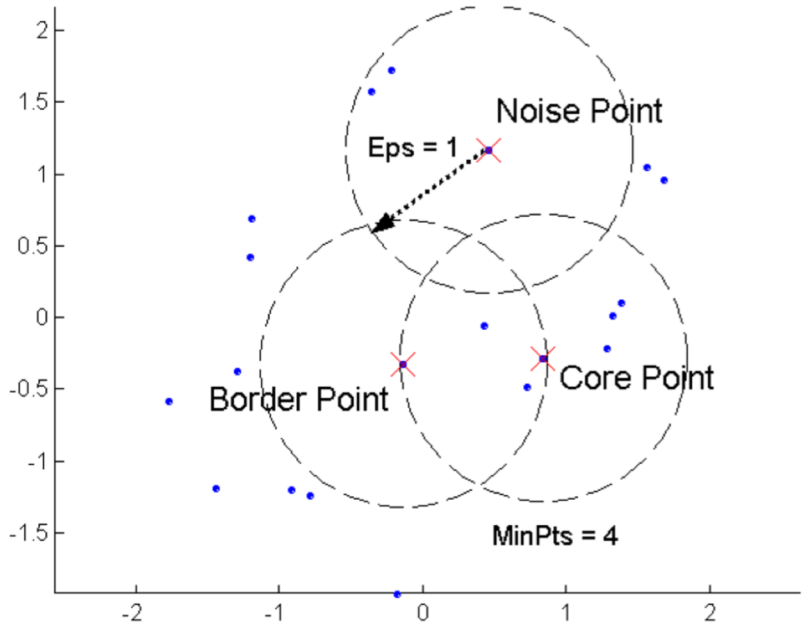




Main idea: divide **noise** (low-density points) from **objects** to clusters (dense points)

DBSCAN

- **Density** = number of points within a specified radius (Eps)
- **Core point** = point with at least a specified number of points (MinPts) within Eps
- **Border point** = a point in the neighborhood of a core point
- **Noise point** = any point that is not a core or a border point

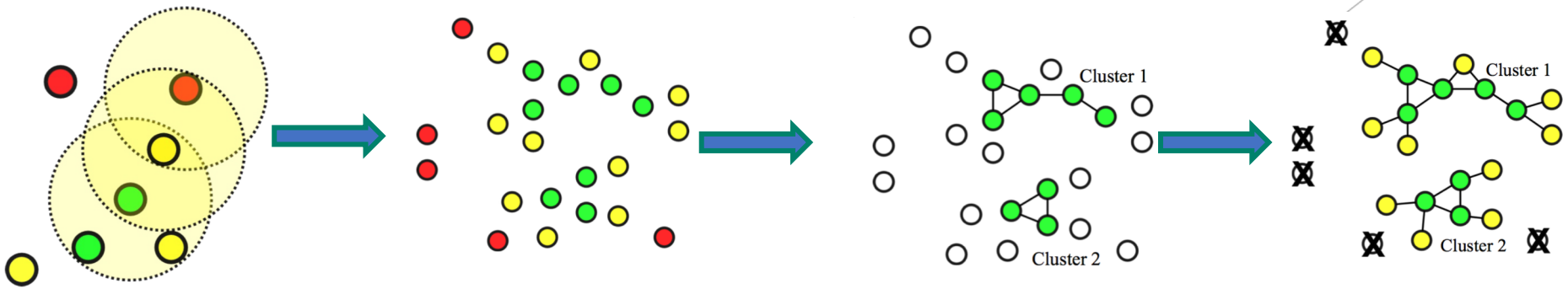


DBSCAN Algorithm

- **Border points can be neighbors of several clusters→ arbitrarily choose one!**

Algorithm 8.4 DBSCAN algorithm.

1: Label all points as core, border, or noise points.
2: Eliminate noise points.
3: Put an edge between all core points that are within *Eps* of each other.
4: Make each group of connected core points into a separate cluster.
5: Assign each border point to one of the clusters of its associated core points.



Point labeling

Connect core objects that are neighbors

Noise elimination

Core object

Border object

Noise

Hierarchical Clustering

Agglomerative

- Start with the points as individual clusters
- At each step, merge the closest pair of clusters until only one cluster (or k clusters) left

Divisive

- Start with one, all-inclusive cluster
- At each step, split a cluster until each cluster contains an individual point (or there are k clusters)

Agglomerative Clustering Algorithm

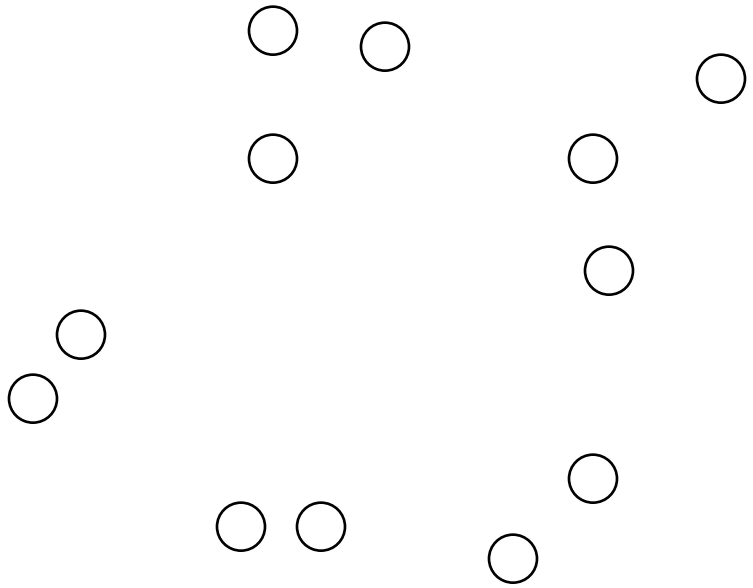
- **Algorithm**
  1. Compute the proximity matrix
  2. Let each data point be a cluster
  3. Repeat:
     - Merge the two closest clusters
     - Update the proximity matrix
  4. Until only a single cluster remains

- Key element is the notion of proximity of two clusters

- Different proximity methods lead to different clusterings

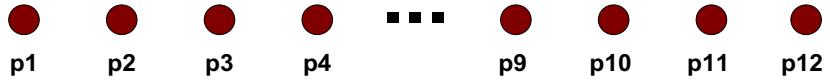
Initial state

- Each cluster is a singleton
- The proximity matrix contains **proximity between single points**



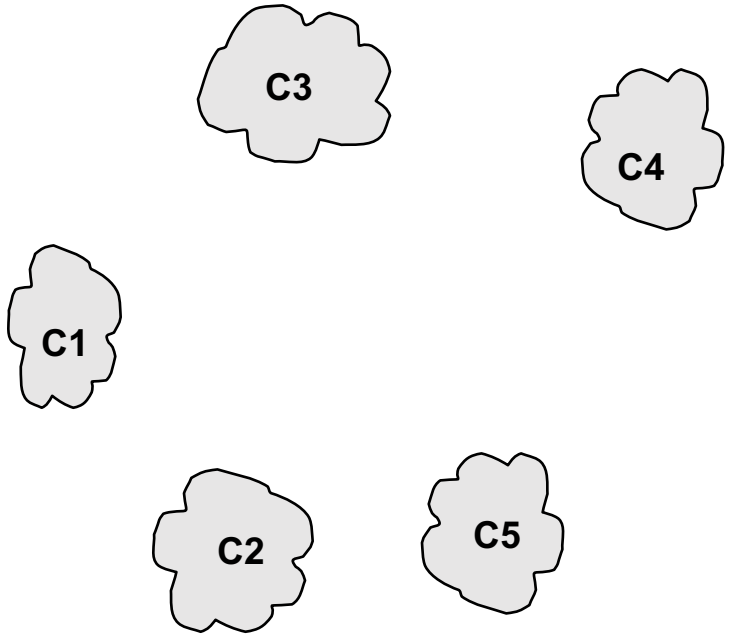
	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix



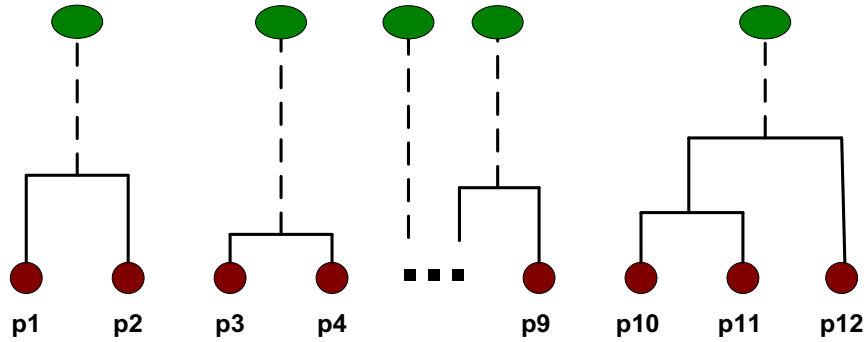
Intermediate State

- After some merging steps, we have some clusters containing more points
- Proximity matrix describes **proximity between groups of points**



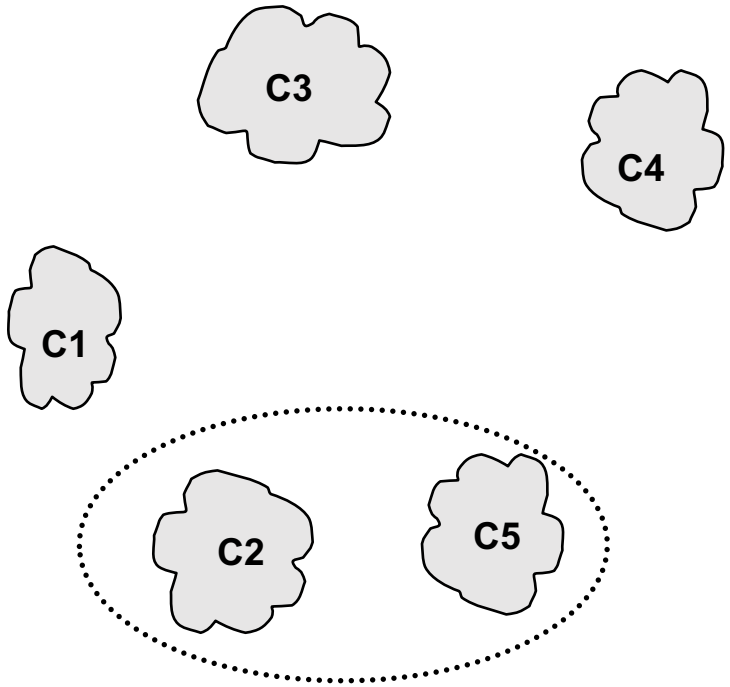
	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix



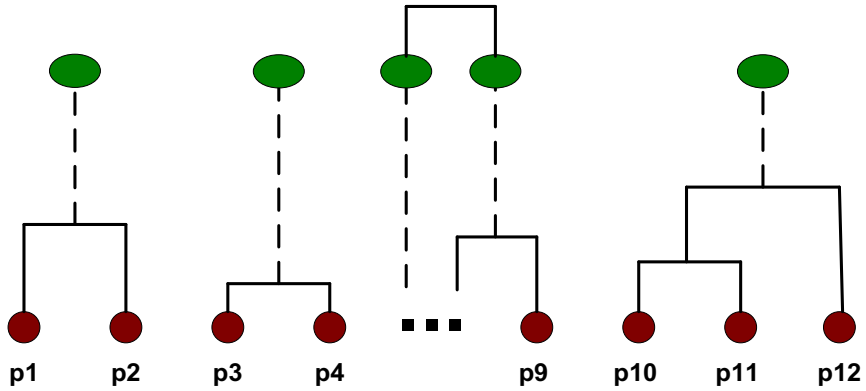
Intermediate state

- Ex: Merge the two closest clusters C2 and C5 and update the proximity matrix



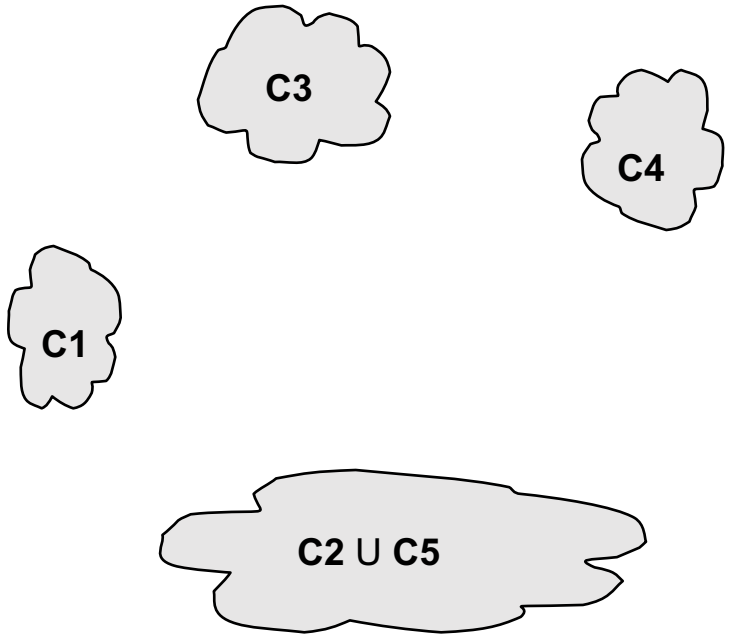
	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix



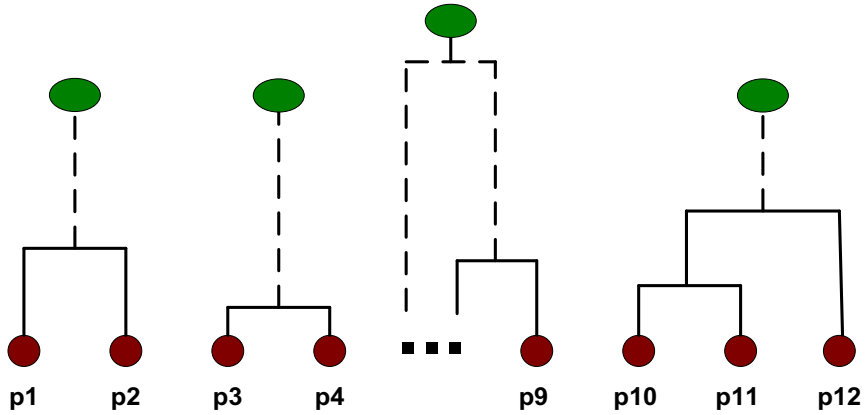
After Merging

- How to update the proximity matrix?



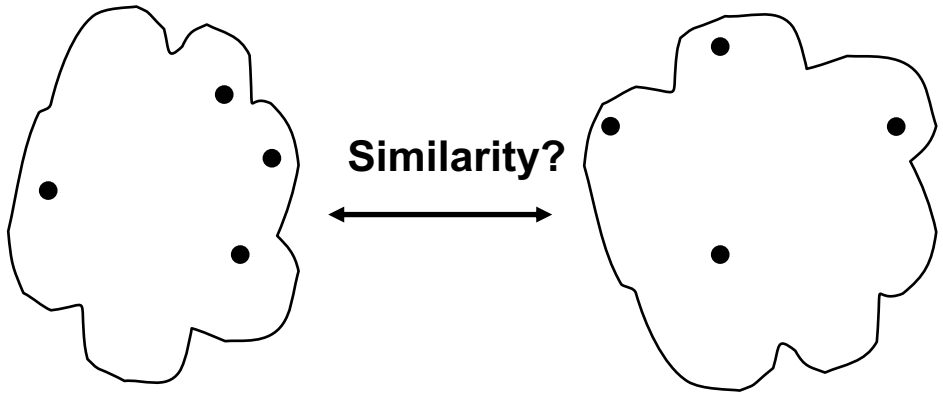
	C1	C2 U C5	C3	C4
C1		?		
C2 U C5	?	?	?	?
C3		?		
C4		?		

Proximity Matrix



Dendrogram

Inter-Cluster Proximity

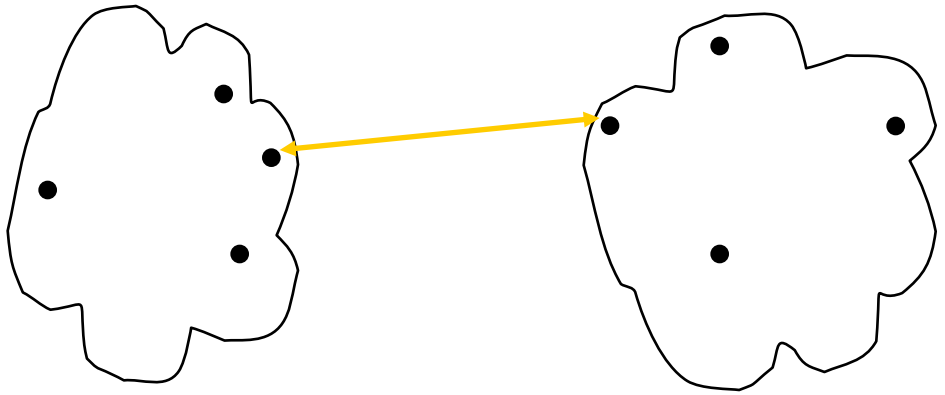


- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

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

Proximity Matrix

Inter-Cluster Proximity

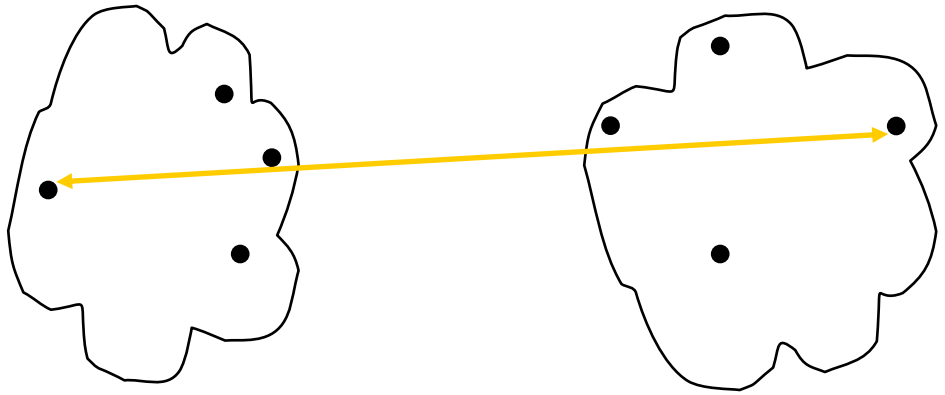


- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

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

Proximity Matrix

Inter-Cluster Proximity

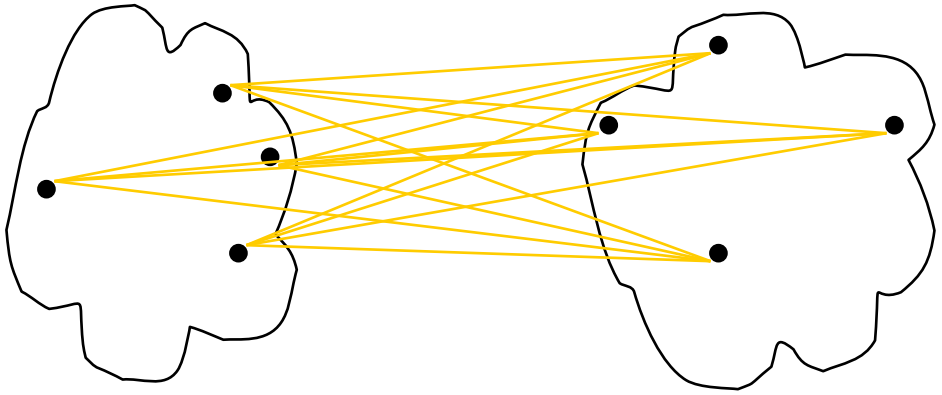


- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

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

Proximity Matrix

Inter-Cluster Proximity

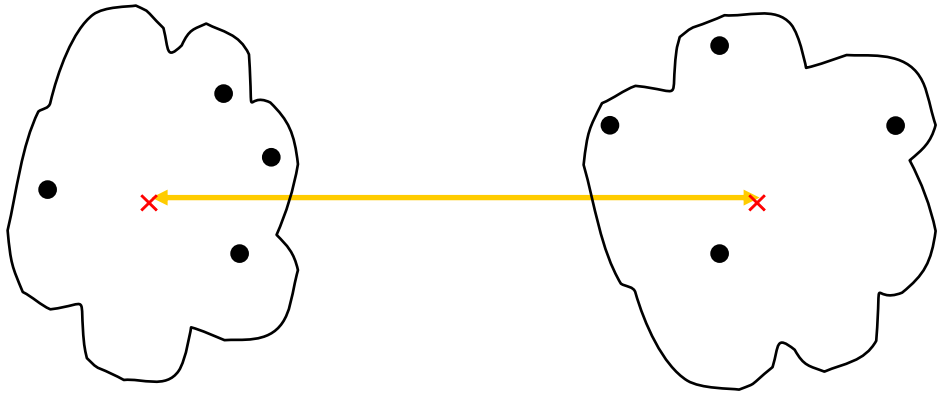


- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

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

Proximity Matrix

Inter-Cluster Proximity



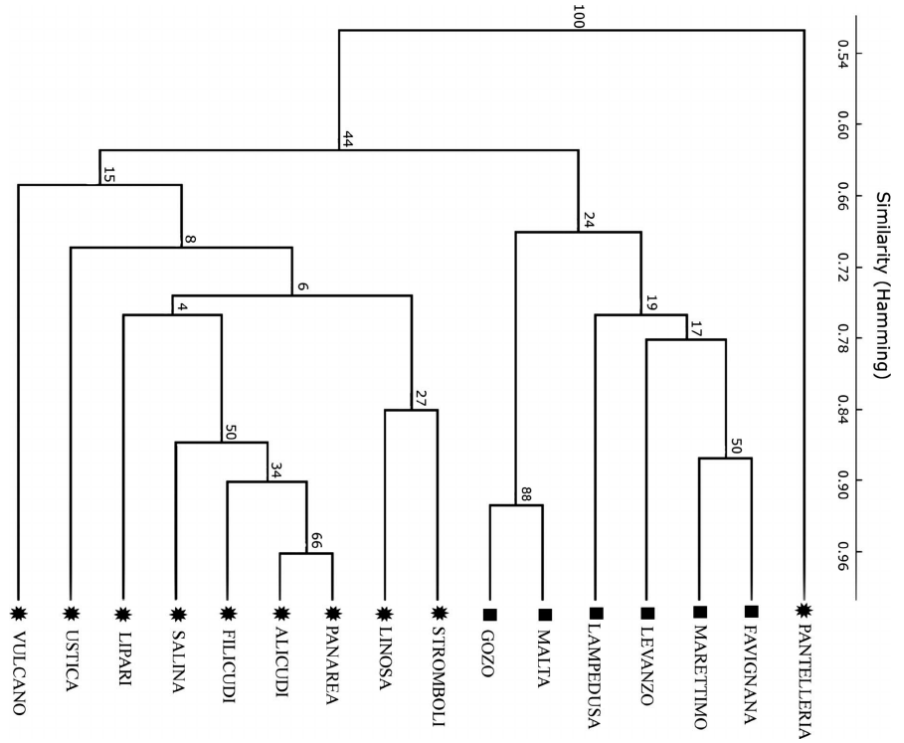
- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward's Method uses squared error

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

Proximity Matrix

Advantages & LIMITATIONS

- **Any assumption about the number of clusters**
  - Any desired number of clusters can be obtained by ‘cutting’ the dendrogram at the proper level
- **Clusters may correspond to meaningful taxonomies**
- **Computational cost high due to proximity matrix computation at any merging step**



References

- The content of these slides is mainly based on the book

- **Introduction to Data Mining, 2nd Edition**
- Pang-Ning Tan, Michigan State University
- Michael Steinbach, University of Minnesota
- Anuj Karpatne, University of Minnesota
- Vipin Kumar, University of Minnesota