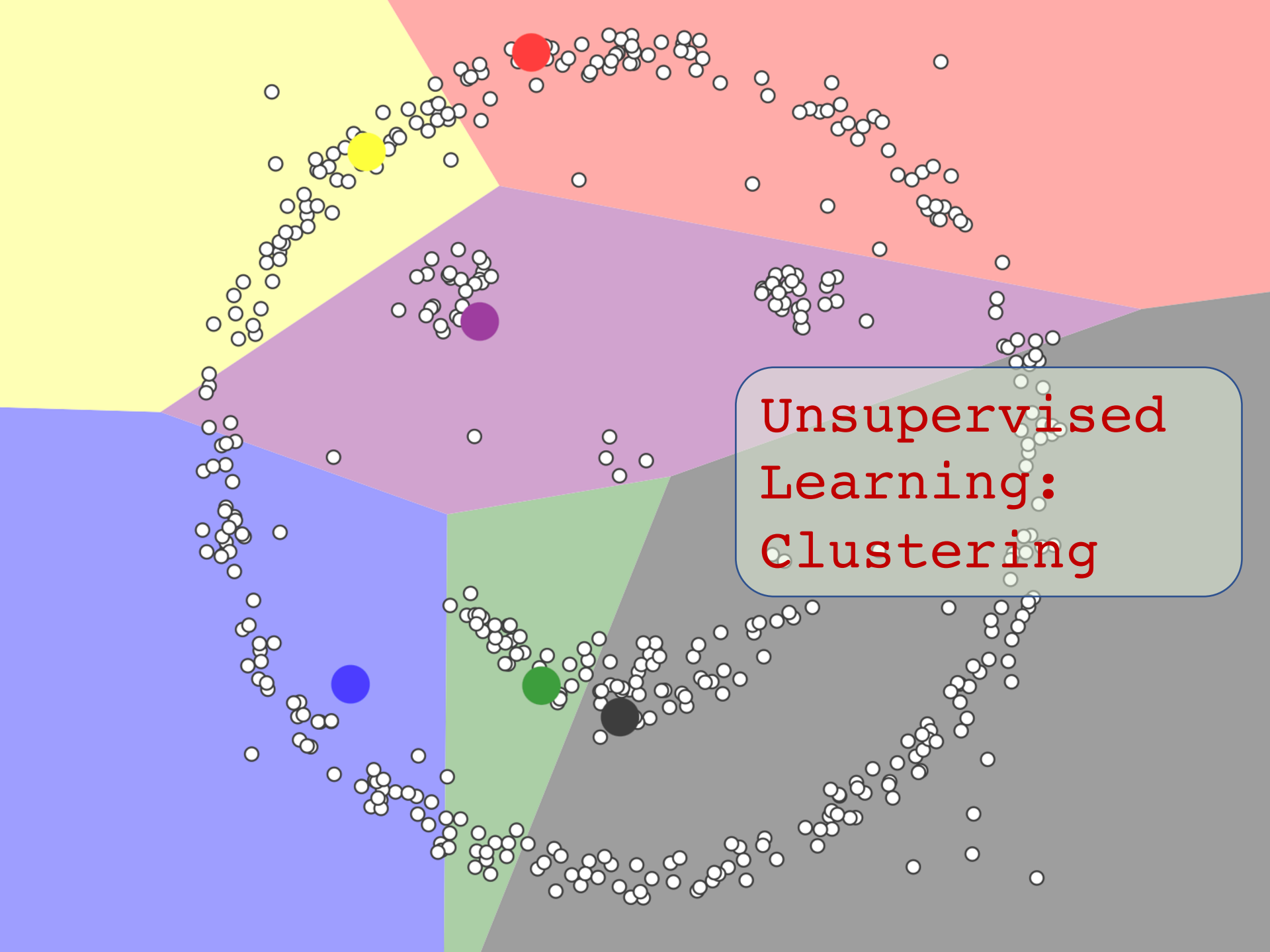

Unsupervised
Learning:
Clustering

Roadmap

- Clustering: Data Analytics Perspective
  - Concepts and Applications
- Data Similarity
- Clustering Approaches
  - K-mean clustering
  - Hierarchical clustering
- Take-home messages

Clustering:
Data Analytics Perspective

What is a Cluster?
Cluster Analysis? Clustering?

- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Grouping a set of data objects into clusters
- Clustering is an *unsupervised learning approach*, *with* no predefined classes
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

Example: Face clusters

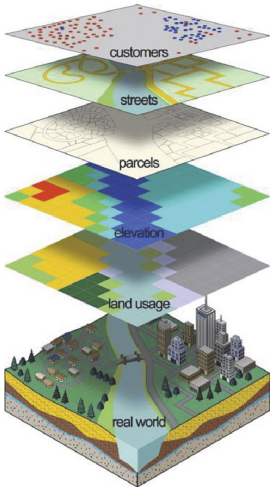
High intra-cluster similarity!

Low inter-cluster similarity!

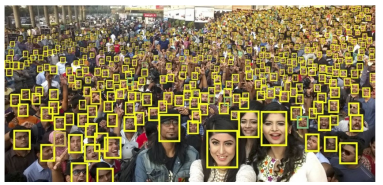
Key concept: Similarity or distance

Similarity of faces! …
Similarity of stocks! …
Similarity of behavior! …Etc.

General Applications of Clustering



- Spatial Data Analysis
  - create thematic maps in GIS by clustering feature spaces
  - detect spatial clusters and explain them in spatial data mining
- Image Processing (cf. face detection via clustering of skin color pixels)



- Economic Science (especially market research; grouping of customers)
- WWW
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns

What is Good Clustering?

- A good clustering method will produce high quality clusters with
  - high intra-class similarity
  - low inter-class similarity
- The quality of a clustering result depends on both the similarity measure used by the method and its implementation.
- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns.

Requirements of Clustering

- Scalability (e.g. large scale clustering of webpages)
- Ability to deal with different types of attributes
- Discovery of clusters with arbitrary shape (see this Demo)
- Minimal requirements for domain knowledge to determine input parameters
- Able to deal with noise and outliers (see previous demo)
- Insensitive to order of input records
- High dimensionality
- Incorporation of user-specified constraints
- Interpretability and usability

Data Similarity

Data Structures

- Data matrix
  - object-by-variable structure (n objects & p variables/ attributes)

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

- Dissimilarity matrix
  - object-by-object structure
  - n objects here!

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix}$$

Measure the Quality of Clustering

- Dissimilarity/Similarity metric: Similarity is expressed in terms of a distance function $d(i, j)$
- There is a separate "quality" function that measures the "goodness" of a cluster.
- The definitions of distance functions are usually very different for interval-scaled, boolean (binary), categorical, and ordinal variables.
- Weights should be associated with different variables based on applications and data semantics.
- It is hard to define "similar enough" or "good enough"
  - the answer is typically highly subjective.

Similarity and Dissimilarity Between Objects

- Distances are normally used to measure the similarity or dissimilarity between two data objects
- Some popular ones include: *Minkowski distance*:

$$d(\vec{i},\vec{j}) = \sqrt[q]{(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \ldots + |x_{ip} - x_{jp}|^q)}$$ L_q norm

where $\vec{i} = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $\vec{j} = (x_{j1}, x_{j2}, \ldots, x_{jp})$ are two *p*-dimensional data objects, and *q* is a positive integer

- If $q=1$, d is Manhattan (or city block) distance

$$d(\vec{i},\vec{j}) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots + |x_{ip} - x_{jp}|$$ L_1 norm

Similarity and Dissimilarity Between Objects (cont.)

- *If* $q=2$, d is Euclidean distance:

$$d(\vec{i},\vec{j}) = \sqrt{(|x_{i_1} - x_{j_1}|^2 + |x_{i_2} - x_{j_2}|^2 + \ldots + |x_{i_p} - x_{j_p}|^2)}$$

L_2 norm

  - Properties
    - $d(i,j) \geq 0$
    - $d(i,i) = 0$
    - $d(i,j) = d(j,i)$
    - $d(i,j) \leq d(i,k) + d(k,j)$

Remember this in geometry?

$x^2 + y^2 = z^2$ *or* $z = \sqrt{x^2 + y^2}$

- Also, one can use weighted distance, parametric Pearson product moment correlation, or other disimilarity measures.

Dissimilarity between Binary Variables

- Example

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

Convert to

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	1	1	0	1	0	0	0
Mary	0	1	0	1	0	1	0
Jim	1	1	1	0	0	0	0

$d(Jack, Mary) = |1-0| + |1-1| + |0-0| + |1-1| + |0-0| + |0-1| + |0-0| = 2$

$d(Jack, Jim) = |1-1| + |1-1| + |0-1| + |1-0| + |0-0| + |0-0| + |0-0| = 2$

$d(Mary, Jim) = |0-1| + |1-1| + |0-1| + |1-0| + |0-0| + |1-0| + |0-0| = 4$

Typically, the distance value is normalized to [0, 1] by dividing it by the total number of attributes, i.e. 7 in this example.

Distance Measure for Nominal/Categorical Variables

- A generalization of the binary variable in that it can take more than 2 states, e.g., red, yellow, blue, green
- Method 1: Simple matching
  - *m*: # of matches, *p*: total # of variables $$d(i,j) = \frac{p-m}{p}$$
- Method 2: use a large number of binary variables
  - creating a new binary variable for each of the *M* nominal states
  - 1-hot encoding

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

Clustering Approaches

Major Clustering Approaches

- Partitioning algorithms: Construct various partitions and then evaluate them by some criteria
- Hierarchical algorithms: Create a hierarchical decomposition of the set of data (or objects) using some criteria

These two are most well-known in general applications

- Density-based: based on connectivity and density functions (see this Demo)
- Grid-based: based on a multiple-level granularity structure
- Model-based: A model is hypothesized for each of the clusters and the idea is to find the best fit of that model to each other

Partitioning Algorithms: Basic Concept

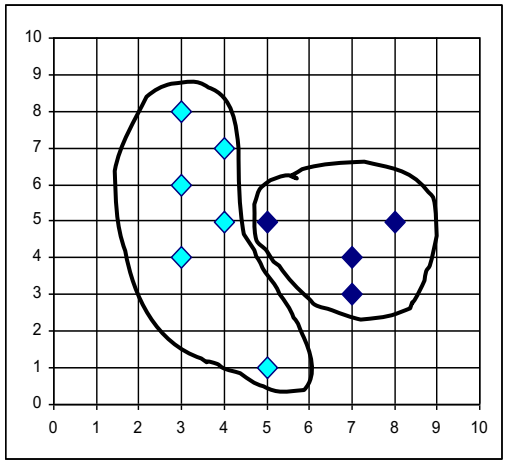
- *Partitioning approach:*
  - Construct a partition of a database ***D*** of ***n*** objects into a set of ***k*** clusters
- Given a particular ***k***, find a partition of *k clusters* that optimizes the chosen partitioning criterion (e.g. high intra-class similarity)
- Two methods
  - Globally optimal method: exhaustively enumerate all partitions (nearly impossible for large *n*)
  - Heuristic methods: *k-means* and *k-medoids* algorithms
    - *k-means* (MacQueen'67): Each cluster is represented by the center of the cluster
    - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

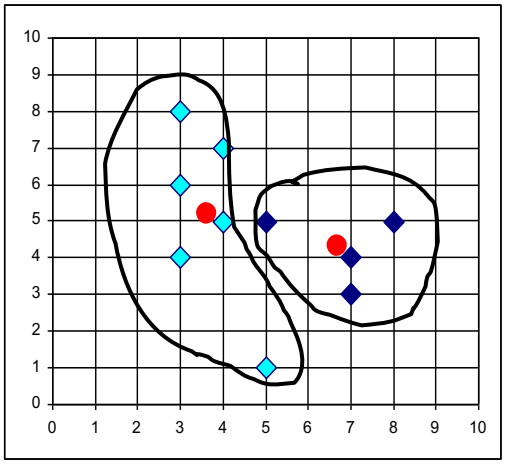
K-Means Clustering Method

- Given *k*, the *k-means* algorithm can be implemented by these four steps:
  1. *Initialization: Partition objects into k nonempty subsets*
  2. *Mean-op: Compute seed points as the centroids of the clusters of the current partition. The centroid is the center (mean point) of the cluster.*
  3. *Nearest_Centroid-op: Assign each object to the cluster with the nearest seed point.*
  4. *Go back to the step 2, stop when no more new assignment.*

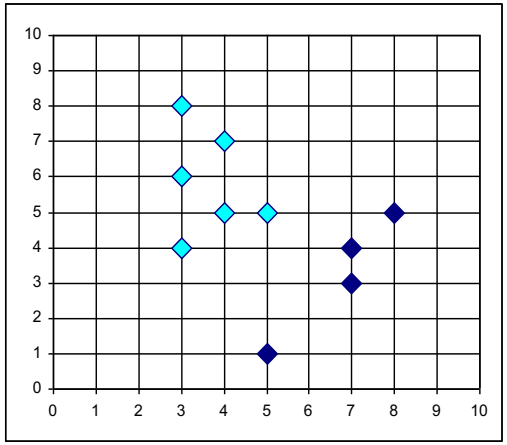
K-Means Clustering Method (see demo here)

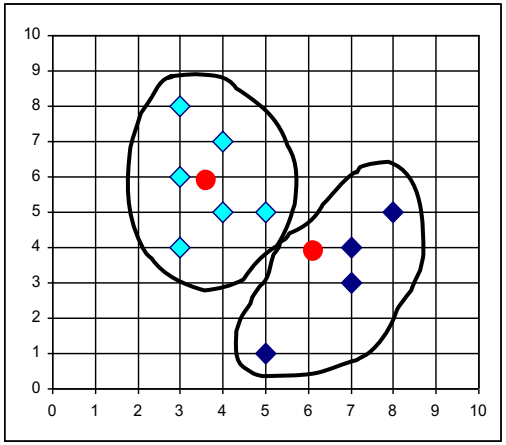
- Example





Red dots denote the cluster centroids





Comments on the *K-Means* Method

- *Strength*
  - *Relatively efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, k, $t << n$.
  - Often terminates at a *local optimum*. The *global optimum* may be found using techniques such as: *deterministic annealing* and *genetic algorithms*
- *Weakness*
  - Applicable only when *mean* is defined, then what about categorical data? What is the mean of red, orange and blue?
  - 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; the basic cluster shape is spherical (convex shape)*

K-means Clustering: ML Perspective

k-Means Clustering

- Find *k* reference vectors (prototypes/codebook vectors/codewords) which best represent data $\mathbf{x}^t \; \forall t$
- Reference vectors, m_j, $j = 1,\ldots,k$
- Use nearest (most similar) reference:

$$\left\|\mathbf{x}^t - \mathbf{m}_i\right\| = \min_j \left\|\mathbf{x}^t - \mathbf{m}_j\right\|$$

- Reconstruction error

$$E\left(\{\mathbf{m}_i\}_{i=1}^k | \mathcal{X}\right) = \sum_t \sum_i b_i^t \left\|\mathbf{x}^t - \mathbf{m}_i\right\|$$

$$b_i^t = \begin{cases} 1 & \text{if } \left\|\mathbf{x}^t - \mathbf{m}_i\right\| = \min_j \left\|\mathbf{x}^t - \mathbf{m}_j\right\| \\ 0 & \text{otherwise} \end{cases}$$

k-Means as an Optimization Problem

Obviously, the goal is to minimize the error *E* or loss L

$$Loss = \sum_{i=1}^{k} \sum_{x_j \in C_i} (x_j - m_i)^2 \quad where\ C_i\ denotes\ ith\ cluster$$

Taking partial derivative of the loss function w.r.t. each cluster prototype m_i and setting it to 0, we have

$$\frac{\partial Loss}{\partial m_i} = \sum_{i=1}^{k} \sum_{x_j \in C_i} \frac{\partial}{\partial m_i} (x_j - m_i)^2 = \sum_{x_j \in Ci} -2(x_j - m_i)$$

Setting $\frac{\partial Loss}{\partial m_i} = \sum_{x_j \in Ci} -2(x_j - m_i) = 0$, we have

$$m_i = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j$$ ← i.e. the mean operation!

k-means Clustering

Initialize $\boldsymbol{m}_i, i = 1, \ldots, k$, for example, to k random $\boldsymbol{x}^t$

Repeat

For all $\boldsymbol{x}^t \in \mathcal{X}$

$$b_i^t \leftarrow \begin{cases} 1 & \text{if } \|\boldsymbol{x}^t - \boldsymbol{m}_i\| = \min_j \|\boldsymbol{x}^t - \boldsymbol{m}_j\| \\ 0 & \text{otherwise} \end{cases}$$

Nearest centroid

For all $\boldsymbol{m}_i, i = 1, \ldots, k$

$$\boldsymbol{m}_i \leftarrow \sum_t b_i^t \boldsymbol{x}^t / \sum_t b_i^t$$

Mean operation

Until $\boldsymbol{m}_i$ converge

Choosing *k*

- Defined by the application, e.g., image quantization or customer segmentation
- Plot data (after dimensionality reduction (to be taught later)) and check for clusters
- Incremental algorithm: Add one at a time until “elbow” (i.e. a sudden increase/decrease of a measure)
- Manually check for meaning

After Clustering

- Clustering methods find similarities between instances and group instances
- Allows knowledge extraction through
  - number of clusters,
  - prior probabilities,
  - cluster parameters, i.e., center, range of features.

Example: Customer segmentation

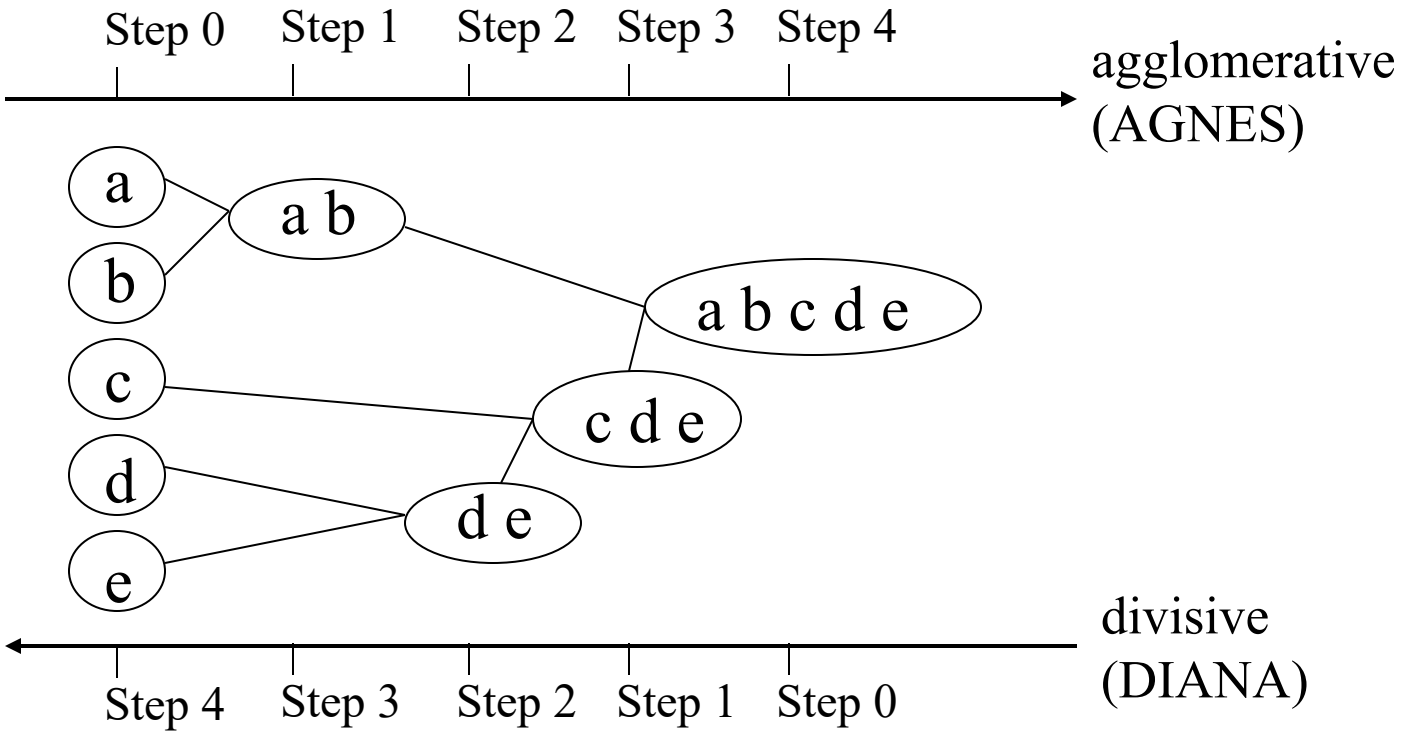
Hierarchical Clustering: Algorithmic Perspective

Hierarchical Clustering Methods

- The clustering process involves a series of partitioning of the data
  - It may run from a single cluster containing all records to n clusters each containing a single record.
- Two popular approaches
  - Agglomerative (ANGES) & divisive (DIANA) methods
- The results may be represented by a dendrogram
  - Diagram illustrating the fusions or divisions made at each successive stage of the analysis.

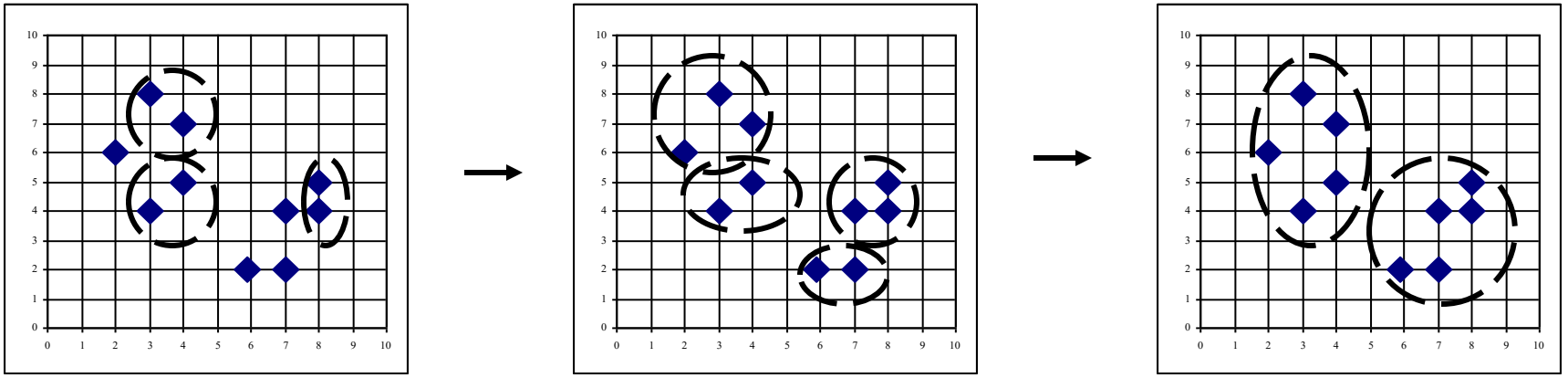
Hierarchical Clustering

- Use distance matrix as clustering criteria. This method does not require the number of clusters ***k*** as an input, but needs a termination condition



AGNES (Agglomerative Nesting)

- Introduced in Kaufmann and Rousseeuw (1990)
- Implemented in statistical analysis packages, e.g. S-Plus
- Use the Single-Linkage method and the dissimilarity matrix.
- Merge nodes that have the least dissimilarity
- Go on in a non-descending fashion
- Eventually all nodes belong to the same cluster



Agglomerative Nesting/Clustering: Single Linkage Method

Basic operations:

START:

- Each cluster of $\{C_1, \cdots, C_j, \cdots, C_n\}$ contains a single individual.

Step 1.

- Find nearest pair of distinct clusters C_i & C_j
- Merge C_i & C_j.
- Decrement the number of cluster by one.

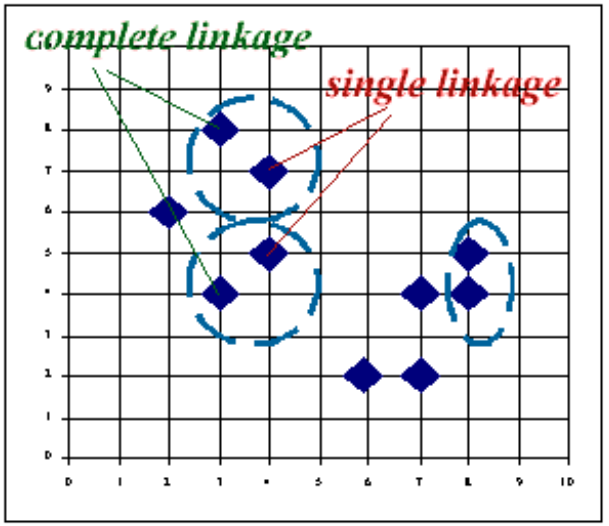
Step 2.

- If the number of clusters equals to one then stop, else return to 1.

Single linkage clustering

- Also known as nearest neighbor (1NN) technique.
- The distance between groups is defined as the closest pair of records from each group.

Agglomerative Nesting/Clustering: Complete Linkage and others



- Complete linkage clustering
  - Also known as furthest neighbor technique.
  - Distance between groups is now defined as that of the most distant pair of individuals (opposite to single linkage method).
- Group-average clustering
  - Distance between two clusters is defined as the average of the distances between all pairs of individuals between the two clusters.
- Centroid clustering
  - Groups once formed are represented by the mean values computed for each attribute (i.e. a mean vector).
  - Inter-group distance is now defined in terms of distance between two such mean vectors.

Single Linkage Method: An Example

- Assume the distance matrix D_1.
- The smallest entry in the matrix is that for individuals 1 and 2, consequently these are joined to form a two-member cluster. Distances between this cluster and the other three individuals are recomputed as
  - $d(12)3 = \min[d13, d23] = d23 = 5.0$
  - $d(12)4 = \min[d14, d24] = d24 = 9.0$
  - $d(12)5 = \min[d15, d25] = d25 = 8.0$
- A new matrix D_2 may now be constructed whose entries are inter-individual distances and cluster-individual values.

$D_1 =$

	1	2	3	4	5
1	0.0				
2	2.0	0.0			
3	6.0	5.0	0.0		
4	10.0	9.0	4.0	0.0	
5	9.0	8.0	5.0	3.0	0.0

$D_2 =$

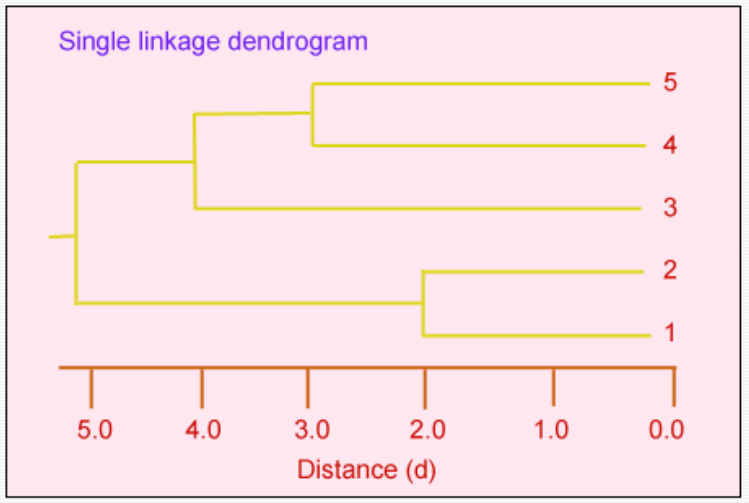
	(12)	3	4	5
(12)	0.0			
3	5.0	0.0		
4	9.0	4.0	0.0	
5	8.0	5.0	3.0	0.0

Single Linkage Method: An Example (cont.)

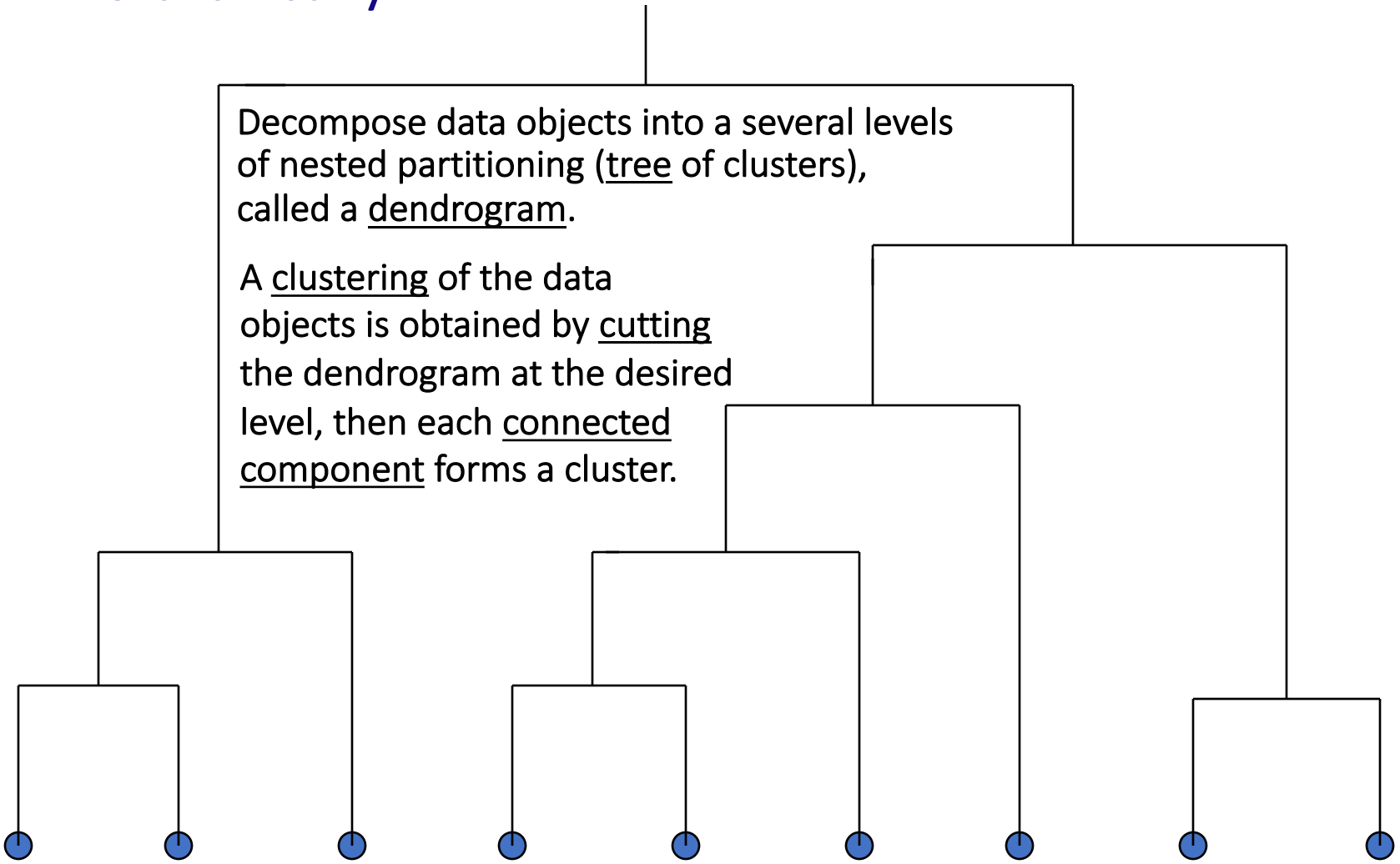
- The smallest entry in D_2 is that for individuals 4 and 5, so these now form a second two-member cluster, and a new set of distances found
  - $d(12)3 = 5.0$ as before
  - $d(12)(45) = \min[d14, d15, d24, d25] = 8.0$
  - $d(45)3 = \min[d34, d35] = d34 = 4.0$
- These may be arranged in a matrix D_3.
- The smallest entry in now $d(45)3$ and so individual 3 is added to the cluster containing individuals 4 and 5. Finally the groups containing individuals 1, 2 and 3, 4, 5 are combined into a single cluster. The partitions produced at each stage are on the right.

$$D_3 = \begin{array}{c} \\ (12) \\ 3 \\ (45) \end{array} \begin{array}{c} \begin{array}{ccc} (12) & 3 & (45) \end{array} \\ \begin{bmatrix} 0.0 & & \\ 5.0 & 0.0 & \\ 8.0 & 4.0 & 0.0 \end{bmatrix} \end{array}$$

Stage	*Groups*
P_1	*[1],[2],[3],[4],[5]*
P_2	*[1,2],[3],[4],[5]*
P_3	*[1,2],[3],[4,5]*
P_4	*[1,2],[3,4,5]*
P_5	*[1,2,3,4,5]*



A *Dendrogram* Shows How the Clusters are Merged Hierarchically



Major weakness of Agglomerative clustering methods

- Do not scale well: time complexity of at least $O(n^2)$, where n is the number of total objects (need to compute the similarity or dissimilarity of each pair of objects)
- Can never undo what was done previously
- Hierarchical methods are biased towards finding 'spherical' clusters even when the data contain clusters of other shapes.
- Partitions are achieved by 'cutting' a dendrogram or selecting one of the solutions in the nested sequence of clusters that comprise the hierarchy.
- Deciding of appropriate number of clusters for the data is difficult.
  - An informal method is to examine the differences between fusion levels in the dendrogram. Large changes are taken to indicate a particular number of clusters

Take-home Messages

- With the class label “disappeared”, the learning problem becomes an unsupervised one.
- A notation of data similarity (or distance) is needed!
- For practitioners, they always struggle with how to compute data similarity! E.g. similarity between hacking activities, similarity between dance movement, etc.
- Clustering is NOT exclusive from classification/regression. It may help classification/regression!

Acknowledgement

- Slides/Materials of
  - Han and Kamber's powerpoint slides for their popular textbook "Data Mining: Concepts and Techniques"
  - E. Alpaydin, Introduction to Machine Learning. 2nd Ed. MIT Press, 2010.
- Photos from Internet