Unsupervised Data Mining: From Batch to Stream Mining Algorithms

Prof. Dr. Stefan Kramer

Johannes Gutenberg-Universität Mainz

Outline

- Introduction to clustering
- Hierarchical clustering

Introduction to Clustering

What is Cluster Analysis?

- *Cluster*: a collection of data objects
  - similar to one another within the same cluster
  - dissimilar to the objects in other clusters
- *Cluster analysis*
  - finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- *Unsupervised* learning: no predefined target class (no teacher)
- Typical applications
  - *stand-alone tool* to get insight into data distribution
  - *preprocessing* for other algorithms

Quality: 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*

Measuring the Quality of Clustering

- Dissimilarity/similarity metric: similarity is expressed in terms of a distance function, typically metric: 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 boolean, nominal, ordinal and continuous variables
  - for variables of mixed type: scale to [0, 1] such that influence of variables is the same
- It is hard to define "similar enough" or "good enough"
  - the answer is typically highly subjective: *human inspection*
  - if clustering is treated as a *density estimation problem*, then it can be *evaluated on test data*!

Requirements of Clustering in Data Mining

- *Scalability*
- Ability to deal with different types of attributes
- Ability to handle *dynamically changing data*
- Discovery of clusters with *arbitrary shape*
- Minimal requirements on domain knowledge to determine input *parameters*
- Able to deal with noise and *outliers*
- Insensitive to order of input records
- *High dimensionality*
- Incorporation of *user-specified constraints*
- Interpretability and usability

Two Types of Data Structures

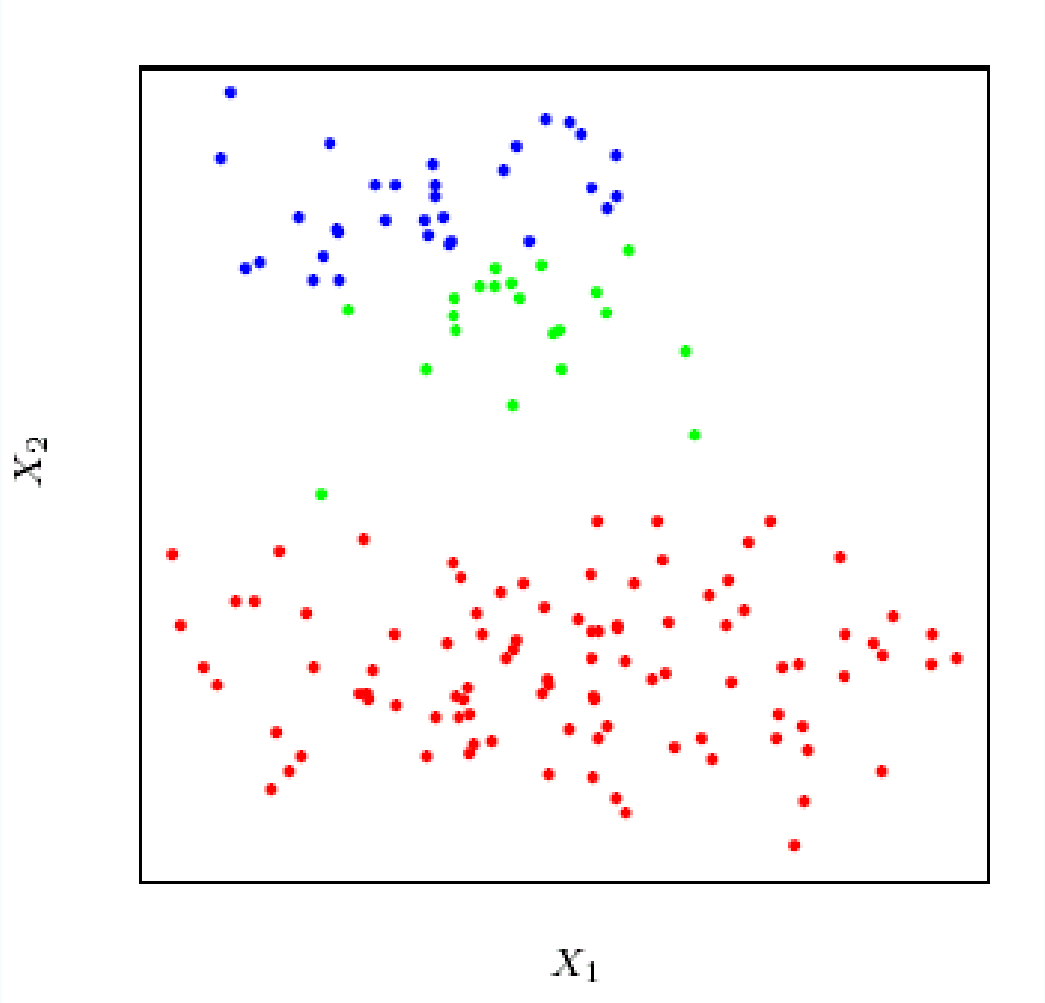
- Data matrix:

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

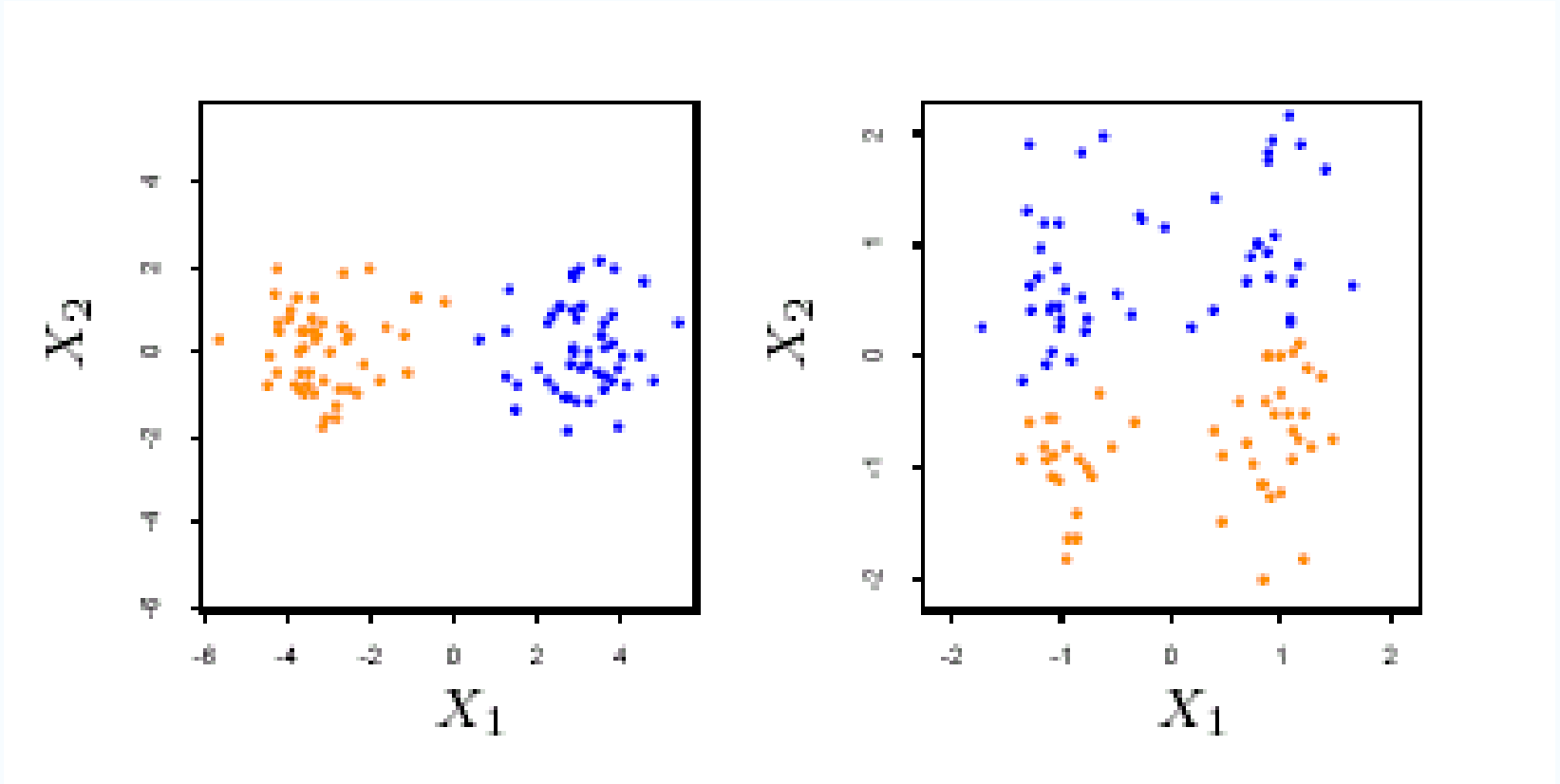
- Dissimilarity/ distance matrix:
  - for some algorithms, only distance matrix is needed

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

Simulated Data Clustered by K-Means Algorithm



Effects of Standardizing on Clustering



Types of *Clusterings*

- Exclusive vs. overlapping
- Categorical vs. probabilistic
- Hierarchical vs. flat
- Online (incremental) vs. batch

Types of *Clustering Methods*

- *Partitioning approach*
  - construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of squared errors
  - in principle no problem: generate all partitions and evaluate
  - typical methods: k-means, k-medoids, CLARANS
- *Hierarchical clustering*
  - create a hierarchical decomposition of the set of data (or objects) using some criterion
  - typical methods: Diana, Agnes, BIRCH, ROCK, CAMELEON
- *Model-based clustering*
  - a *model is hypothesized* for each of the clusters and tries to find the best fit of that model to each other
  - typical methods: EM, SOM, COBWEB

Types of *Clustering Methods*

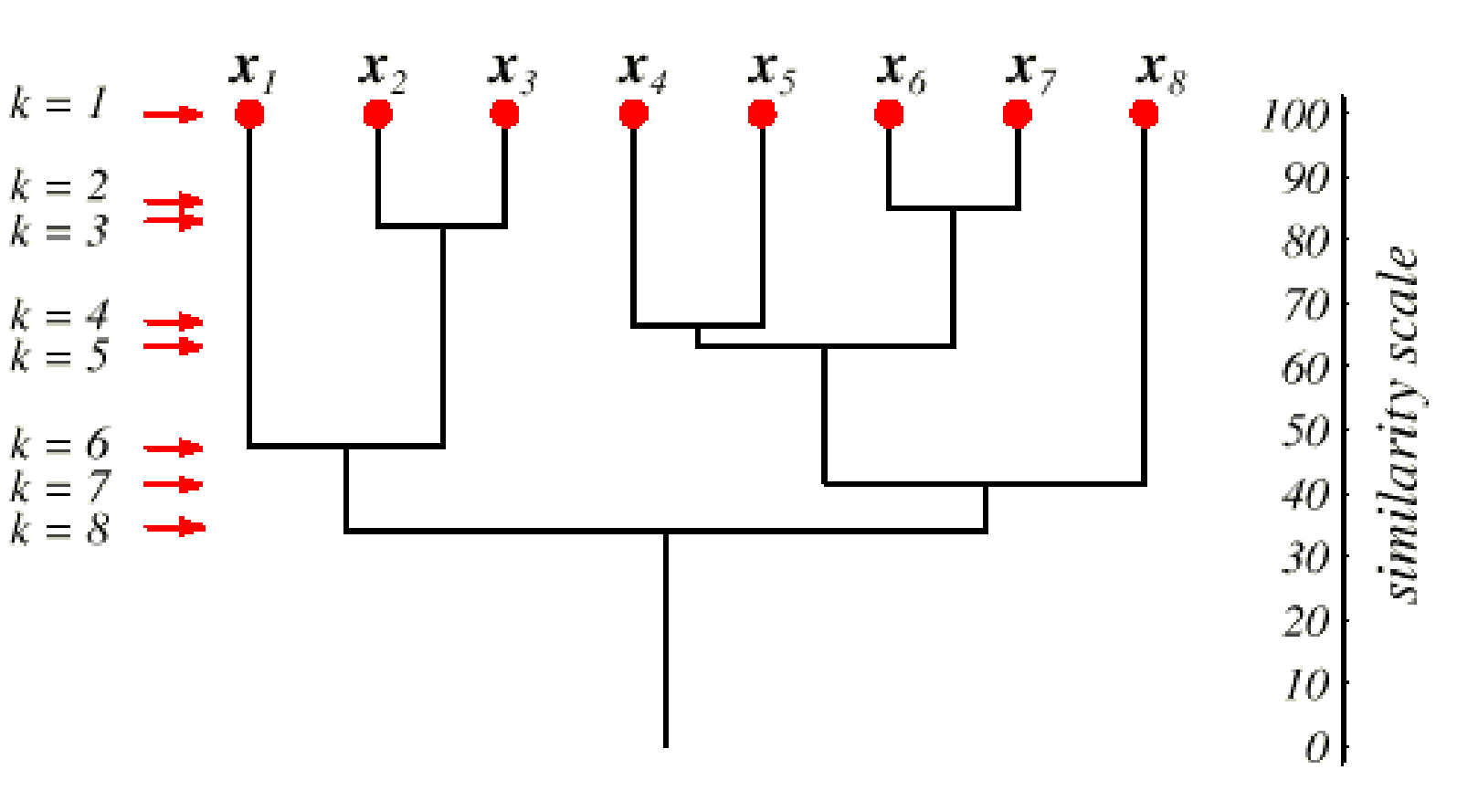
- *Graph-based clustering*
  - typical method: Click
- Special topics / special algorithms:
  - *high-dimensional data*
    - subspace clustering
  - *density-based clustering*
    - based on connectivity and “density functions”
  - *frequent pattern-based clustering*
    - based on the analysis of frequent patterns
  - *constraint-based clustering*
    - clustering by considering user-specified or application-specific constraints

Hierarchical Clustering

Hierarchical Clustering

- *Top down*: find two clusters and then proceed recursively for the two subsets
- *Bottom up*: at each step join the two closest clusters (starting with single-instance clusters)
- Design decision: distance between *clusters*, e.g., two closest instances in clusters vs. distance between means
- Both methods produce a so-called *dendrogram*

Dendrograms



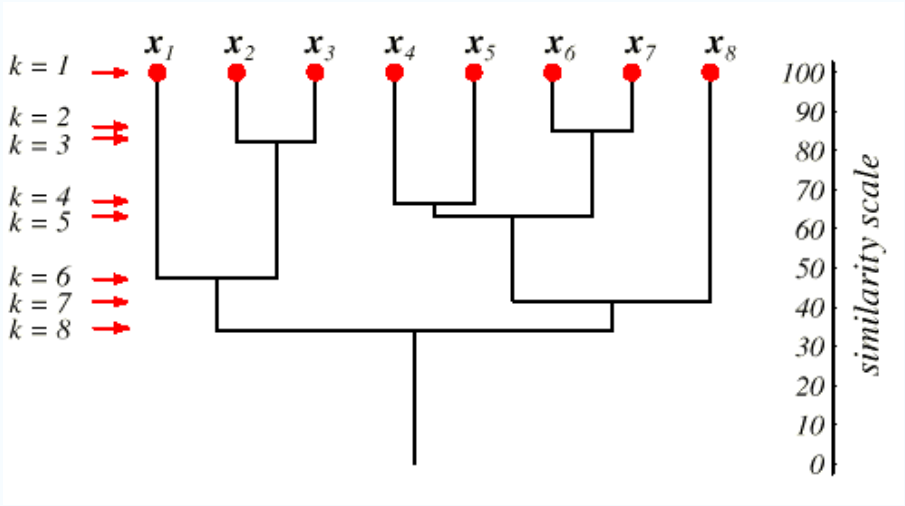
Hierarchical Agglomerative Clustering

```
for i = 1, ..., n let C_i = {x(i)};
while there is more than one cluster left do
    let C_i and C_j be the clusters
        minimizing the distance D(C_k, C_h) between any two clusters;
    C_i = C_i ∪ C_j;
    remove cluster C_j;
end;
```

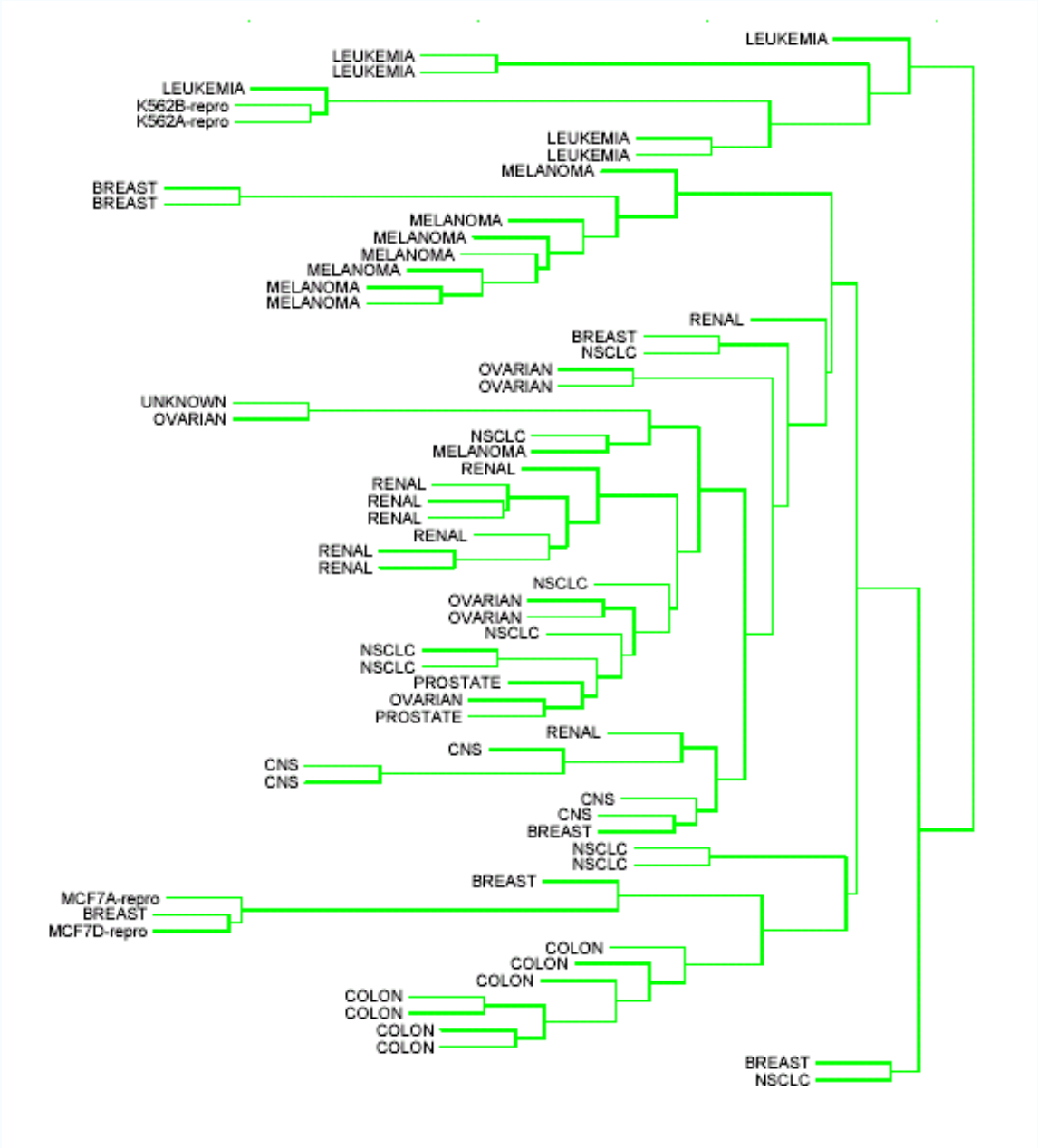
Time complexity of $O(n^3)$ or $O(n^2 \log n)$ if optimized for single linkage/complete linkage using a priority queue

Dendrograms

Given any two samples x and x', they will be grouped together *at some level*, and if they are grouped a level k, they remain grouped for all higher levels



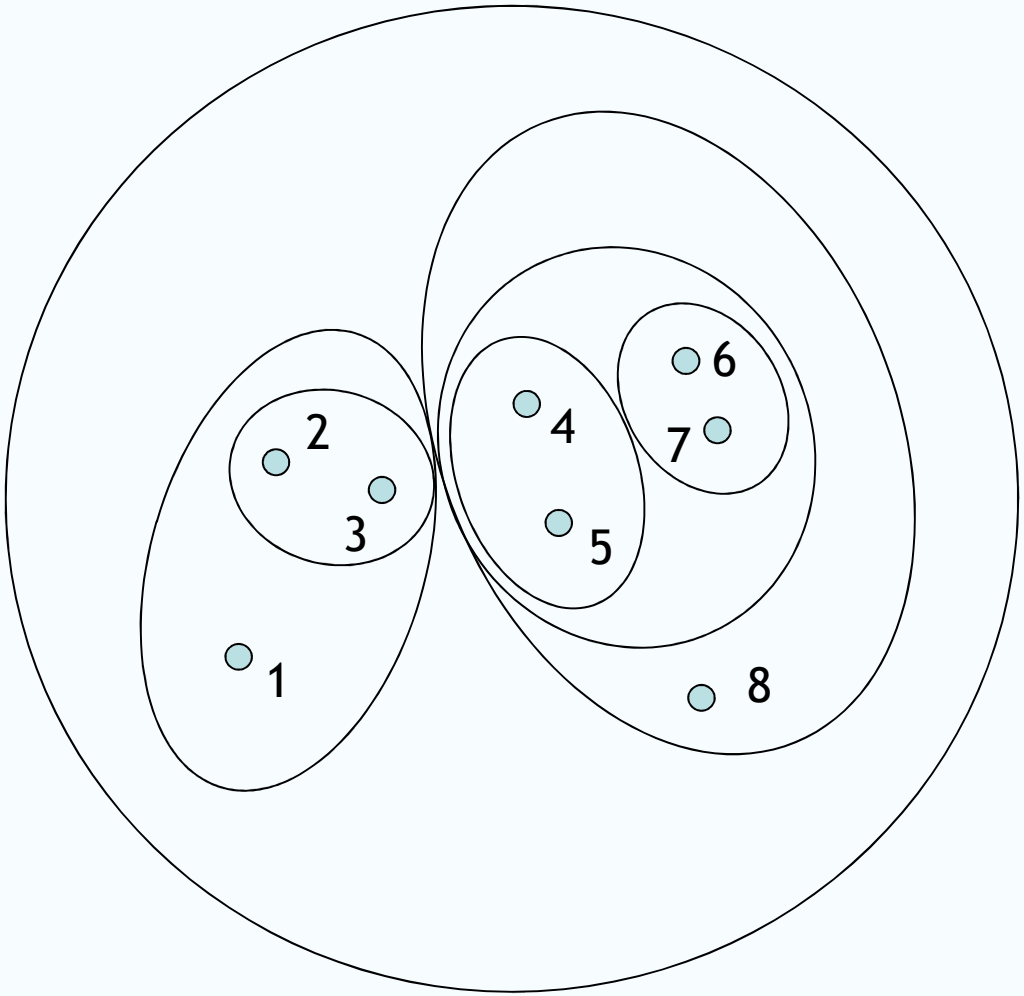
Dendrogram

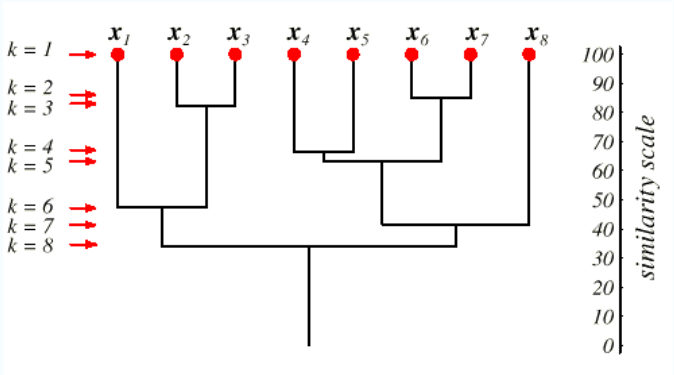


Hierarchical agglomerative clustering (average linkage) applied to human tumor microarray data

Possible only because dissimilarity is monotone increasing with the level of the merger!

Alternative Visualization by Venn Diagram





Note

- *The hierarchical structure is returned whether there exists one in the data or not!*
- A dendrogram is the description of the results of an algorithm, not a graphical summary of the data

Variants of Hierarchical Agglomerative Clustering

- Single linkage

$$d_{SL}(G,H) = \min_{\substack{i \in G, \\ i' \in H}} d_{ii'}$$

- Complete linkage

$$d_{CL}(G,H) = \max_{\substack{i \in G, \\ i' \in H}} d_{ii'}$$

- Group average/average linkage

$$d_{GA}(G,H) = \frac{1}{N_G N_H} \sum_{i \in G} \sum_{i' \in H} d_{ii'}$$

Dissimilarity Measures 1

Def. *Metric*:

(1) Nonnegativity
(2) Reflexivity

$$d_{ij} > d_{ii} = 0$$

(3) Symmetry

$$d_{ij} = d_{ji}$$

(4) Triangle inequality

$$d_{ij} \leq d_{ik} + d_{kj}$$

Dissimilarity Measures 2

(5) "Ultrametric triangle inequality"

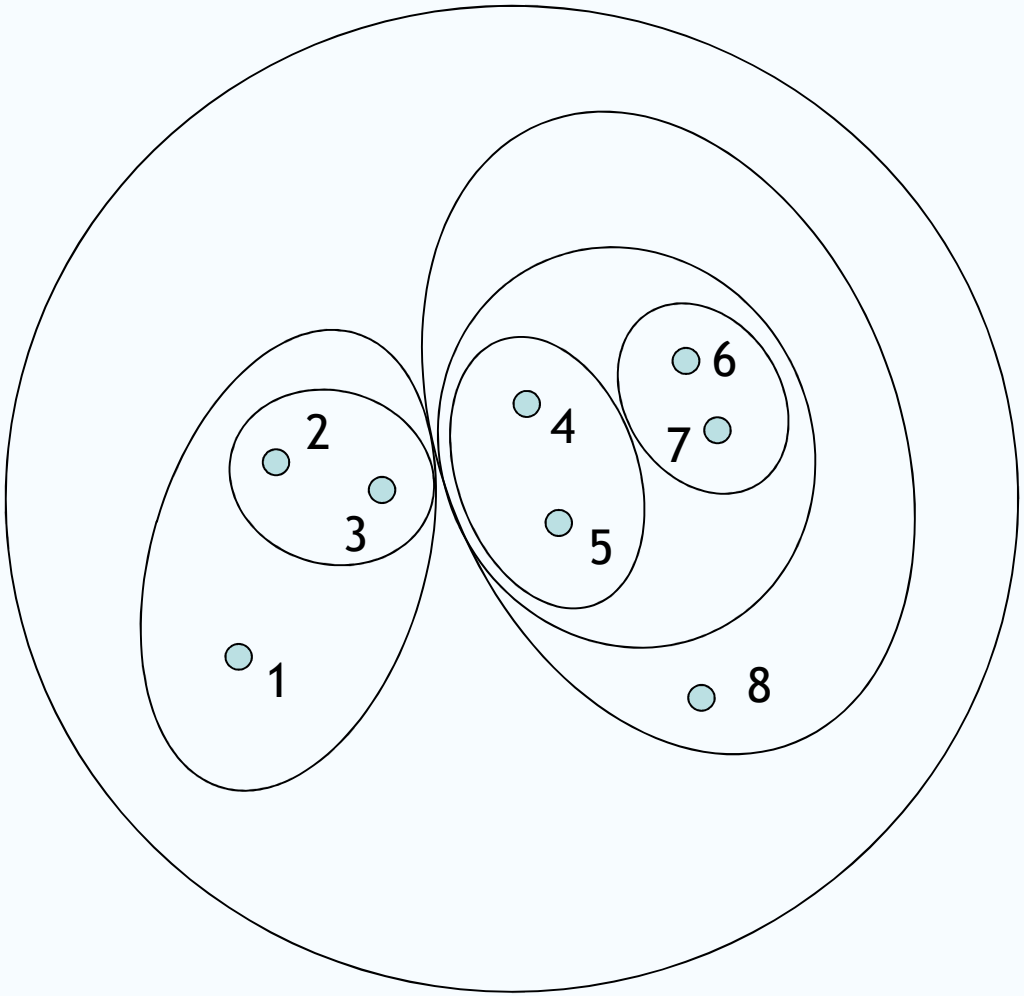
$$d_{ij} \leq \max(d_{ik}, d_{kj})$$

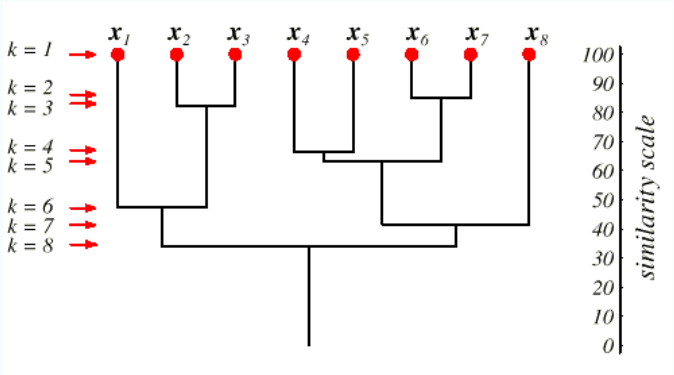
- At least two of $d_{i,j}$, $d_{i,k}$ and $d_{k,j}$ are the same
- *Ultrametric*: (1),(2),(3),(5)

Dissimarity Measures 3

- Agglomerative clustering just requires a dissimilarity measure with (1), (2), (3)
- Given a single linkage or complete linkage dendrogram, we *obtain* an ultrametric by the so-called *cophenetic* dissimilarity
- If the dissimilarity measure is already an ultra-metric, then all three methods compute the same result

Alternative Visualization by Venn Diagram





Properties of Variants 1

- *Compactness* property: observations within clusters should be close together
- *Closeness* property: observations should be closer to members of its own cluster than to members of other clusters

Properties of Variants 2

- ***Single linkage:***
  - „chaining effect“
  - potentially clusters with large *diameter*
  - violating „compactness“ property
- ***Complete linkage:***
  - compact clusters
  - but violating the „closeness“ property

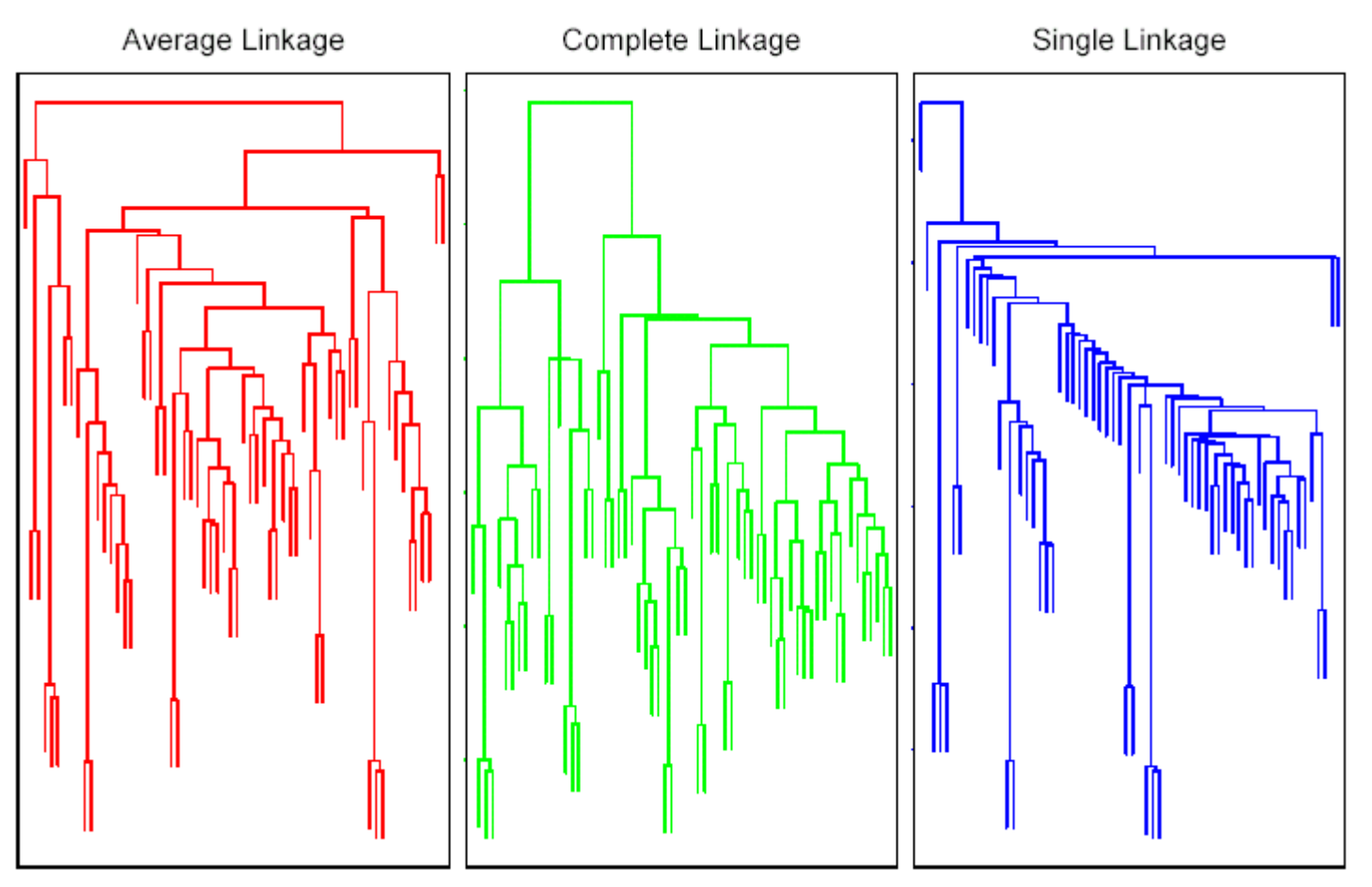
Properties of Variants 3

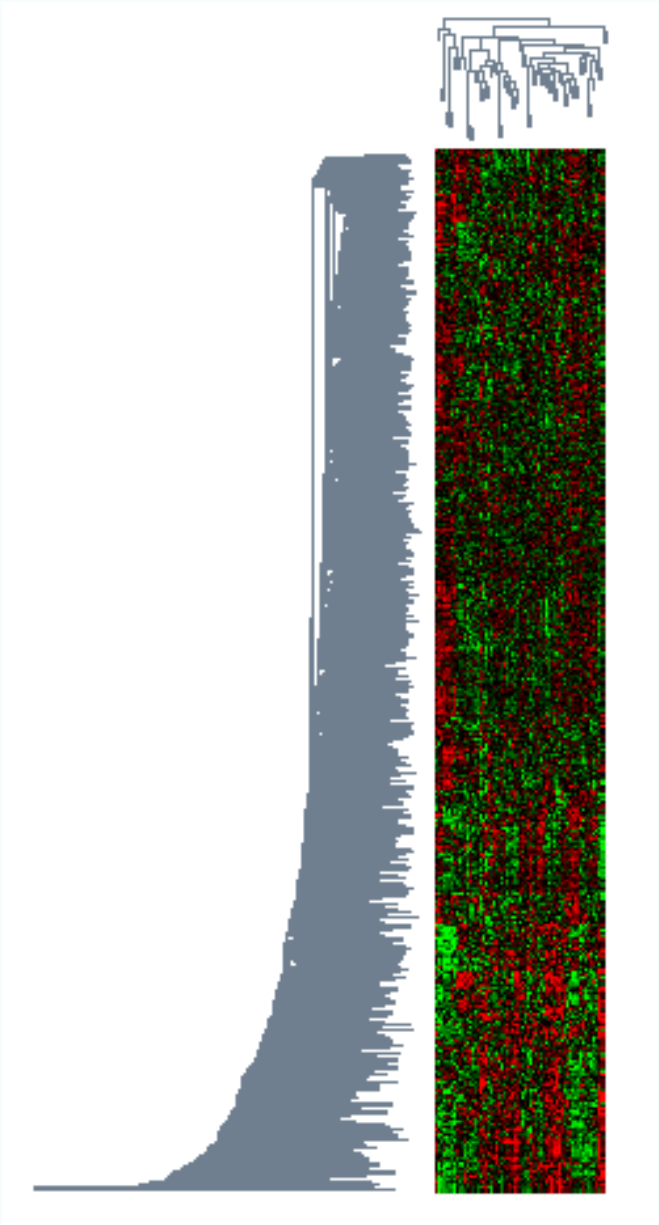
- ***Group average/average linkage:***
  - compromise: relatively compact clusters, relatively far apart
  - estimate of a characteristic of the relationship between two densities of the clusters G and H

$$\int\int d(x,x')p_G(x)p_H(x')dxdx'$$

  - d being the distance between x and x'
  - approaches this characteristic as sample size approaches infinity (what for d_{SL} and d_{CL}?)

Variants of Hierarchical Agglomerative Clustering





Clustering of Rows (Genes) and Columns (Samples) Independently

Ordering both rows and colums

Hierarchical Clustering (Continued)

Divisive Clustering

- Dividing hierarchically in a top-down fashion
- *Monothetic* divisive methods: using one variable at a time
- *Polythetic* divisive methods: making splits on the basis of all variables
- E.g., *Diana*: choose cluster with the largest diameter for splitting
- Divisive clustering: computationally more intensive and less widely used than agglomerative methods

Diana

- One cluster C, compute for each instance in C the average distance from all the other instances:

$$\Delta_{1,x_1} = (n(C) - 1)^{-1} \sum_{y \in C \setminus \{x_1\}} d(x_1, y)$$

Diana

- Pick x_1^* for which $\Delta_{1,x1}$ is maximal
- Repeat picking x_1^*, ..., x_{k+1}^* iteratively until $\Delta_{k+1,x^*_{k+1}} < 0$ and x_{k+1}^* maximizes $\Delta_{k+1,x_{k+1}}$, where

$$\Delta_{k+1,x_{k+1}} = (n(C) - k - 1)^{-1} \sum_{y \in C \setminus \{x_1^*,\ldots,x_k^*\}} d(x_{k+1}, y)$$

$$-k^{-1} \sum_{i=1}^{k} d(x_i^*, x_{k+1}),$$