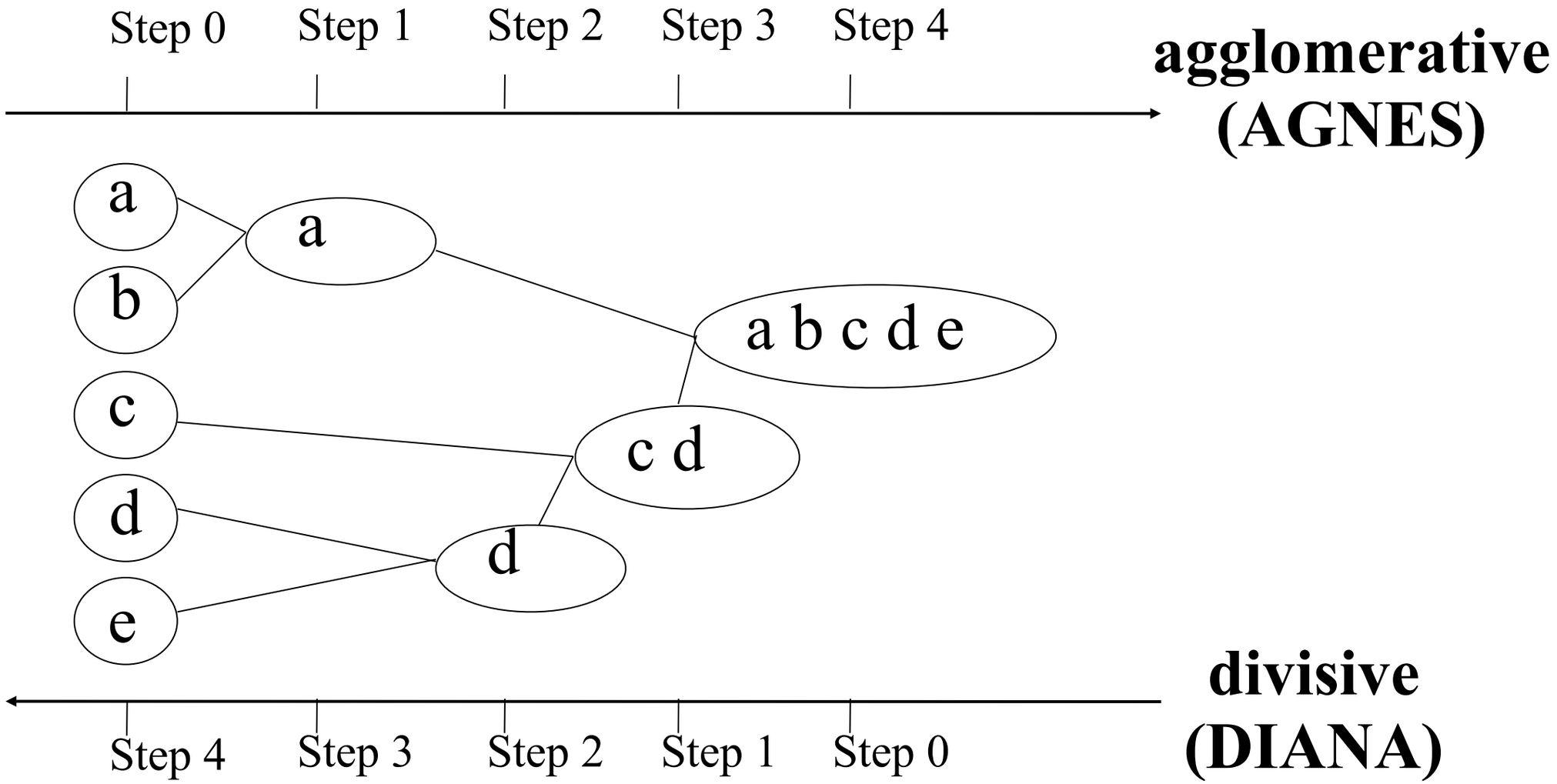
Lecture 25: Clustering 2

Artificial Intelligence
CS-GY-6613-I

Julian Togelius
julian.togelius@nyu.edu

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



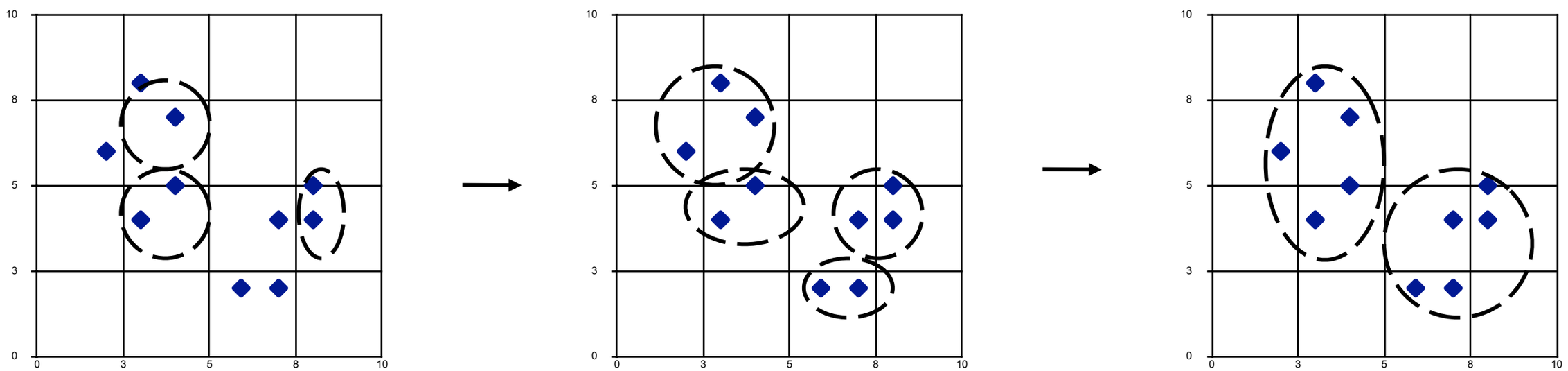
AGNES (Agglomerative Nesting)

- Use the Single-Link method (distance between cluster a and b)=distance between closest members of clusters a and b) 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



Let $X = \{x1, x2, x3, ..., xn\}$ be the set of data points.

1) Begin with the disjoint clustering having level $L(0) = 0$ and sequence number $m = 0$.

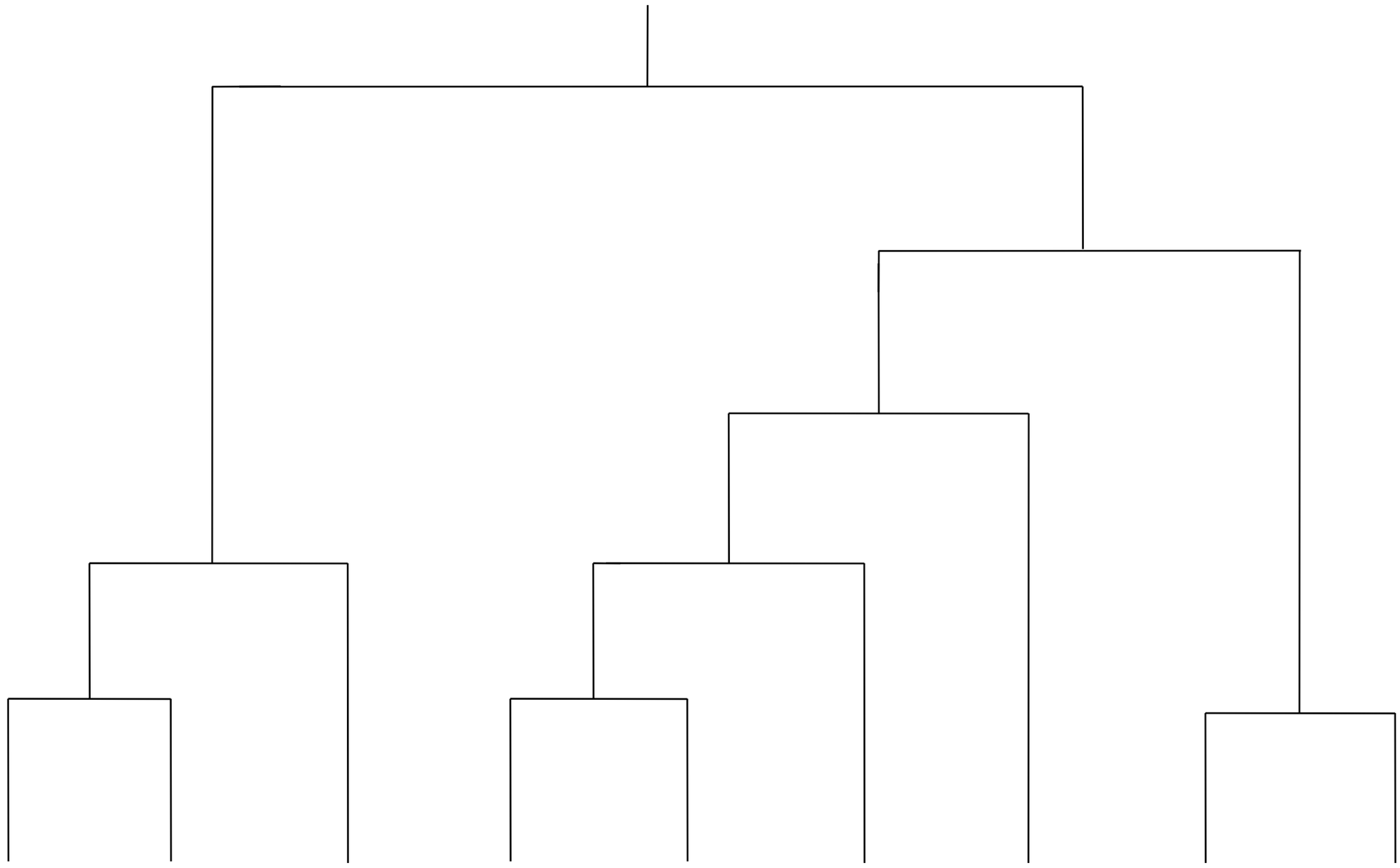
2) Find the least distance pair of clusters in the current clustering, say pair (r), (s), according to $d[(r),(s)] = \min d[(i),(j)]$ where the minimum is over all pairs of clusters in the current clustering.

3) Increment the sequence number: $m = m + 1$. Merge clusters (r) and (s) into a single cluster to form the next clustering m. Set the level of this clustering to $L(m) = d[(r),(s)]$.

4) Update the distance matrix, D, by deleting the rows and columns corresponding to clusters (r) and (s) and adding a row and column corresponding to the newly formed cluster. The distance between the new cluster, denoted (r,s) and old cluster(k) is defined in this way: $d[(k), (r,s)] = \min (d[(k),(r)], d[(k),(s)])$.

5) If all the data points are in one cluster then stop, else repeat from step 2).

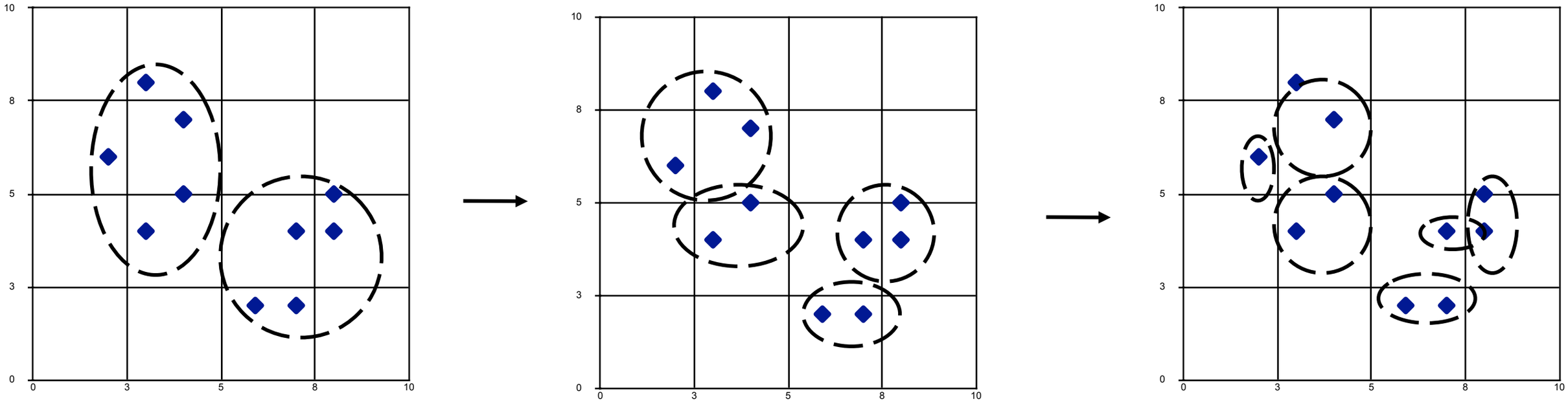
Dendrogram



Where do you cut?

DIANA (Divisive Analysis)

- Inverse order of AGNES
- Eventually each node forms a cluster on its own



High-dimensional data

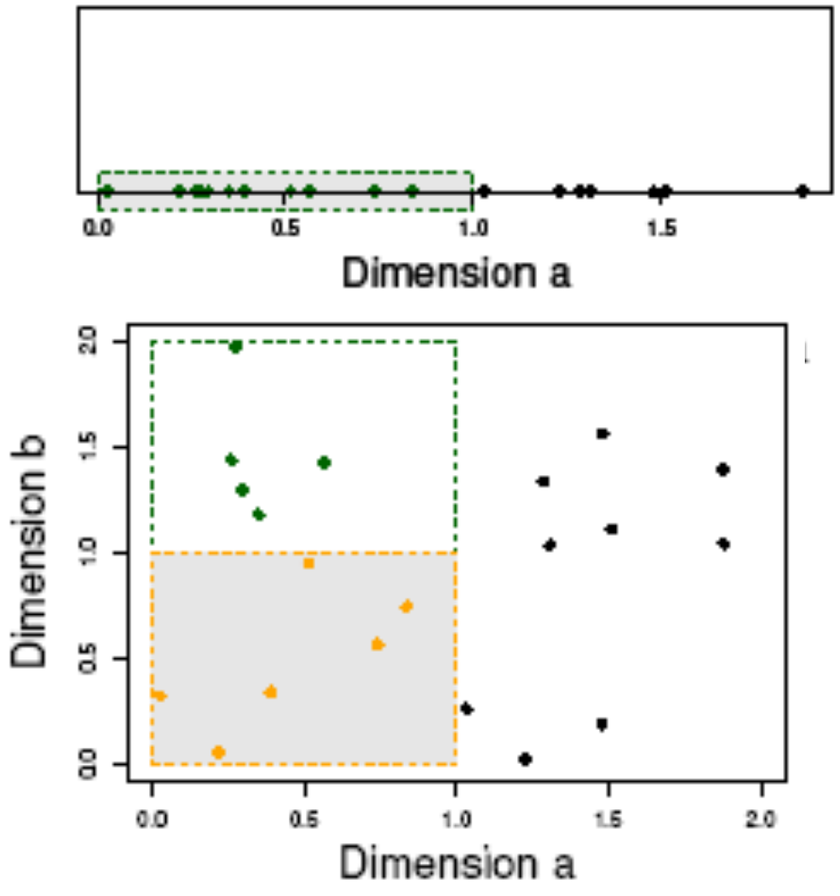
- Clustering high-dimensional data
  - Many applications: text documents, DNA micro-array data
- Major challenges:
  - Many irrelevant dimensions may mask clusters
  - Distance measure becomes meaningless—due to equi-distance
  - Clusters may exist only in some subspaces

High-dimensional data

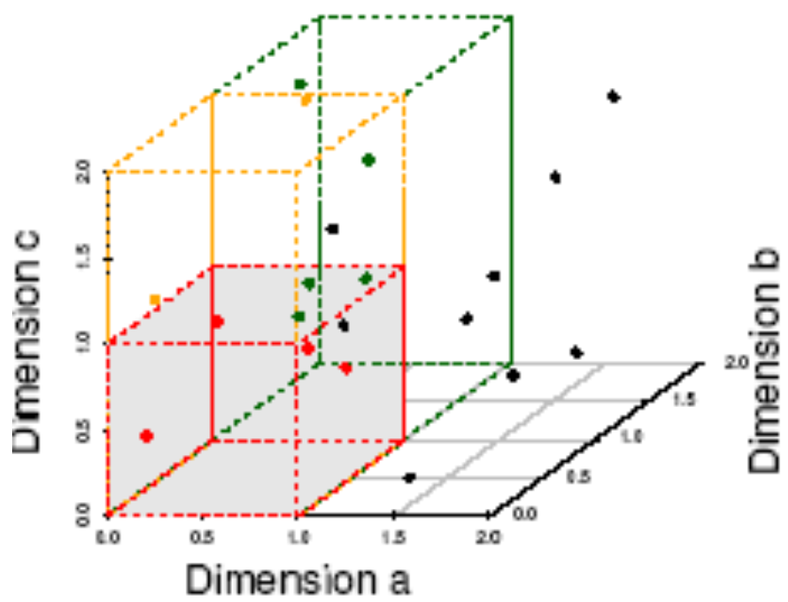
- Methods
  - Feature transformation: only effective if most dimensions are relevant
    - PCA & SVD useful only when features are highly correlated/ redundant
  - Feature selection: wrapper or filter approaches
    - useful to find a subspace where the data have nice clusters
  - Subspace-clustering: find clusters in all the possible subspaces
    - CLIQUE, ProClus, and frequent pattern-based clustering

The curse of dimensionality

- Data in only one dimension is relatively packed
- Adding a dimension “stretches” the points across that dimension, making them further apart
- Adding more dimensions will make the points further apart—high dimensional data is extremely sparse
- Distance measure becomes meaningless—due to equi-distance



(b) 6 Objects in One Unit Bin



(c) 4 Objects in One Unit Bin

Measuring clustering quality

- 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 interval-scaled, boolean, categorical, ordinal ratio, and vector variables.

- It is hard to define “similar enough” or “good enough”; the answer is typically highly subjective.

Silhouette coefficient

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

- Where i is an instance, a(i) is the instance's average similarity to all other points in its own cluster, and b(i) is the similarity of the instance to all points in the closest other cluster

- Silhouette of a clustering: average s(i) of all points