Data-driven Intelligent Systems

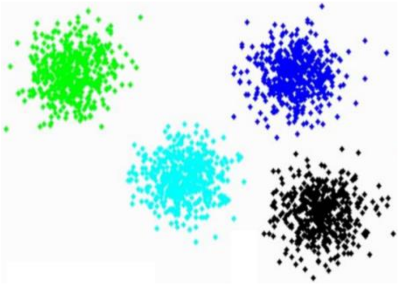
Lecture 19
Clustering

KNOWLEDGE
TECHNOLOGY

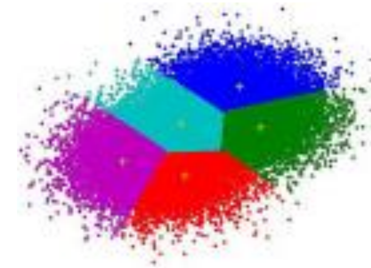
http://www.informatik.uni-hamburg.de/WTM/

Clustering – Overview

- Background of clustering
  - Measure of cluster quality: Davies-Bouldin index
- K-means
- K-medoid
- Hierarchical clustering



What is Cluster Analysis?



- Cluster: A group of data objects
  - similar (or related) to one another within the same group
  - dissimilar (or unrelated) to the objects in other groups
- Cluster analysis
  - *finding similarities* between data according to their characteristics and *grouping* similar data objects into clusters
- ***Unsupervised learning***: no predefined classes
  - the number of clusters may be unknown
- Typical applications
  - ***preprocessing step*** for other algorithms
  - ***stand-alone tool*** to get insight into data distribution

Clustering for Data Understanding and Applications

- Biology: taxonomy of living things: class, family, genus and species
- Information retrieval: document clustering
- Marketing: help marketers discover distinct customer groups, and develop targeted marketing programs
- Land use: similar land use in an earth observation database
- City-planning: identifying groups of houses according to their house type, value (and geographical location)
- Climate: understanding earth climate, find patterns of atmospheric similarities
- Earth-quake studies: observed earth quake epicenters should be clustered along continent faults

Typical Requirements

- Scalability
- High dimensionality
- Ability to deal with different types of attributes
- Incremental clustering and insensitivity to input order
- Ability to deal with noisy data
- Discovery of clusters with arbitrary shape
- Constraint-based clustering
- Domain knowledge to determine input parameters
- Interpretability and usability

Major Clustering Approaches (1)

- ***Partitioning approach***
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of squared errors
  - Typical methods: k-means, k-medoids, CLARANS
- ***Hierarchical approach***
  - Create a hierarchical decomposition of the set of data points
  - Typical methods: Diana, Agnes, BIRCH, CAMELEON
- ***Density-based approach***
  - Based on local connectivity or density above threshold
  - Typical methods: DBSCAN, OPTICS, DenClue
- ***Dimensionality reduction methods***
  - Construct a new space and cluster therein
  - Typical methods: Spectral clustering

Major Clustering Approaches (2)

- ***Grid-based approach***
  - multiple-level granularity structure, finite number of cells
  - Typical methods: STING, WaveCluster, CLIQUE
- ***Model- or Neural Network based***
  - A model is hypothesized and best fitted to each of the clusters
  - Typical: Gaussian Mixture Models, EM, SOM, COBWEB
- ***Frequent pattern-based***
  - Based on the analysis of frequent patterns
  - Typical methods: p-Cluster
- ***Instance-based***
  - Related: *k*-nearest neighbors (kNN) – ***classify*** a data point by the majority vote of its *k* closest neighbour points

Data Matrix and Dissimilarity Matrix

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

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

- ***Data matrix***
  - *n* data points with *p* dimensions each
- ***Dissimilarity matrix***
  - Registers the distances between the *n* data points
  - A triangular *n* × *n* matrix

Reminder: Numeric Data: Minkowski Distance

- ***Minkowski distance***: A popular distance measure

$$d(i, j) = \sqrt[h]{|x_{i1} - x_{j1}|^h + |x_{i2} - x_{j2}|^h + \cdots + |x_{ip} - x_{jp}|^h}$$

  where

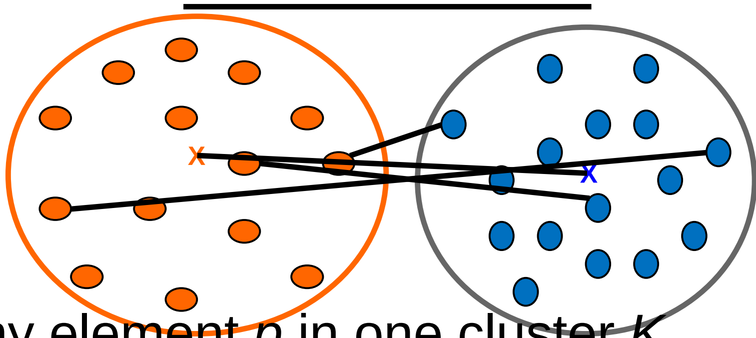
  $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ are two p-dimensional data objects,

  h = order (the distance so defined is also called L-h norm)

- Properties
  - $d(i, j) > 0$ if $i \neq j$, and $d(i, i) = 0$ (Positive definiteness)
  - $d(i, j) = d(j, i)$ (Symmetry)
  - $d(i, j) \leq d(i, k) + d(k, j)$ (Triangle Inequality)

- A distance that satisfies these properties is a **metric**

Distance between Clusters



- Single link: smallest distance between any element *p* in one cluster K_i and any element *q* in the other K^j, i.e., $dist^{SL}(K_i, K_j) = min_{p,q}\ d(x_{ip}, x_{jq})$
- Complete link: largest distance between any element in one cluster and any element in the other, i.e., $dist^{CL}(K_i, K_j) = max_{p,q}\ d(x_{ip}, x_{jq})$
- Average: avg distance between an element in one cluster and an element in the other, i.e., $dist^{avg}(K_i, K_j) = avg_{p,q}\ d(x_{ip}, x_{jq})$
- Centroid: distance between the centroids of two clusters, i.e., $dist^{Cen}(K_i, K_j) = d(C_i, C_j)$
- Medoid: distance between the medoids of two clusters, i.e., $dist^{Med}(K_i, K_j) = d(M_i, M_j)$
  - Medoid: a chosen, centrally located ***object*** in the cluster (whose dissimilarity to all other objects in the cluster is minimal)

Centroid, Radius and Diameter of a Cluster (for numerical data sets)

$$C_i = \frac{\sum_{p=1}^{N_i} x_{ip}}{N_i}$$

- Centroid: the "center of mass" of a cluster (N_i = # points x_{ip} in the cluster i)
  ← vector C_i has minimal average Euclidean distance to all points

$$R_i = \sqrt{\frac{\sum_{p=1}^{N_i} (x_{ip} - C_i)^2}{N_i}}$$

- Radius: standard deviation of the distance of points to centroid of respective cluster *i*

$$D_i = \sqrt{\frac{\sum_{p=1}^{N_i} \sum_{q=1}^{N_i} (x_{ip} - x_{iq})^2}{N_i (N_i - 1)}}$$

- Diameter: standard deviation of the distances between all *pairs* of points in cluster *i*

Clustering – Overview

- Background of clustering
  - Measure of cluster quality: Davies-Bouldin index
- K-means
- K-medoid
- Hierarchical clustering

Quality: What Is Good Clustering?

- The **quality** of a clustering method depends on
  - the ***similarity measure*** used
    - definitions of distance functions vary for interval-scaled, boolean, categorical, or *vector variables* ← *our focus*
  - its ***implementation***, and
  - its ability to discover the ***patterns*** in the data
- High quality clusters:
  - high ***intra-class*** similarity: ***cohesive*** within clusters
  - low ***inter-class*** similarity: ***distinctive*** between clusters
- ***Cluster indices***: <u>Davies-Bouldin</u>, Ray-Turi, Silhouette, …

Measure of Quality: Clustering indices

The **Davies-Bouldin index** *DB* quantifies a clustering result by relating intra- vs. inter-class similarities

$$R_i = \sqrt{\frac{\sum_{p=1}^{N_i} (x_{ip} - C_i)^2}{N_i}}$$

- Intra-class spread (radius):
- Inter-class (centroid) distance: $dist_{ij} = \left\| C_i - C_j \right\|$
- Badness of separation of two clusters *i*, *j*: $D_{ij} = \dfrac{R_i + R_j}{dist_{ij}}$
- For cluster *i*, that other cluster *j* is relevant that is least separated: $D_i^{worst} = \max_{j \neq i} D_{ij}$
- Average over all *k* clusters: $$\boxed{DB = \frac{1}{k} \sum_i^k D_i^{worst}}$$

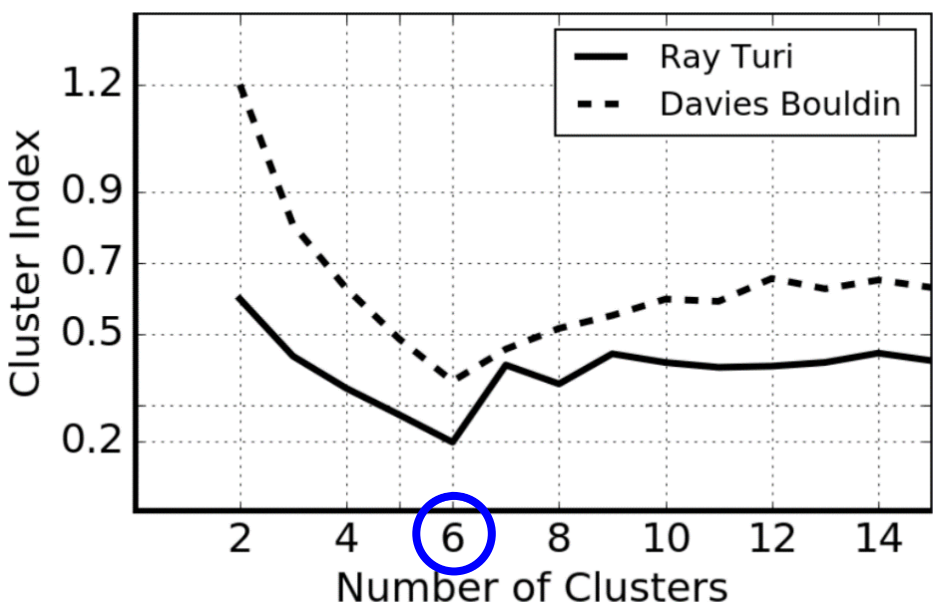
(minimal *DB* is best)

Measure of Quality: Clustering indices

For a typical clustering process

$$D_{ij} = \frac{R_i + R_j}{dist_{ij}}$$

- as the number of clusters *k* goes up
  - intra-class similarity gets higher (good: smaller R_i)
  - inter-class similarity gets higher (bad: smaller $dist_{ij}$)



- Best results empirically found by trying out different *k* and finding the minimum cluster index

Clustering – Overview

- Background of clustering
  - Measure of cluster quality: Davies-Bouldin index
- K-means
- K-medoid
- Hierarchical clustering

Partitioning Algorithms: Basic Concept

- ***Partitioning approach***: partition a database *D* of objects x_p into a set of *k* clusters, minimising sum of squared distances to means μ_i

$$E = \sum_{i=1}^{k} \sum_{p \in C_i}^{N_i} (x_p - \mu_i)^2$$

points assigned to cluster *i*

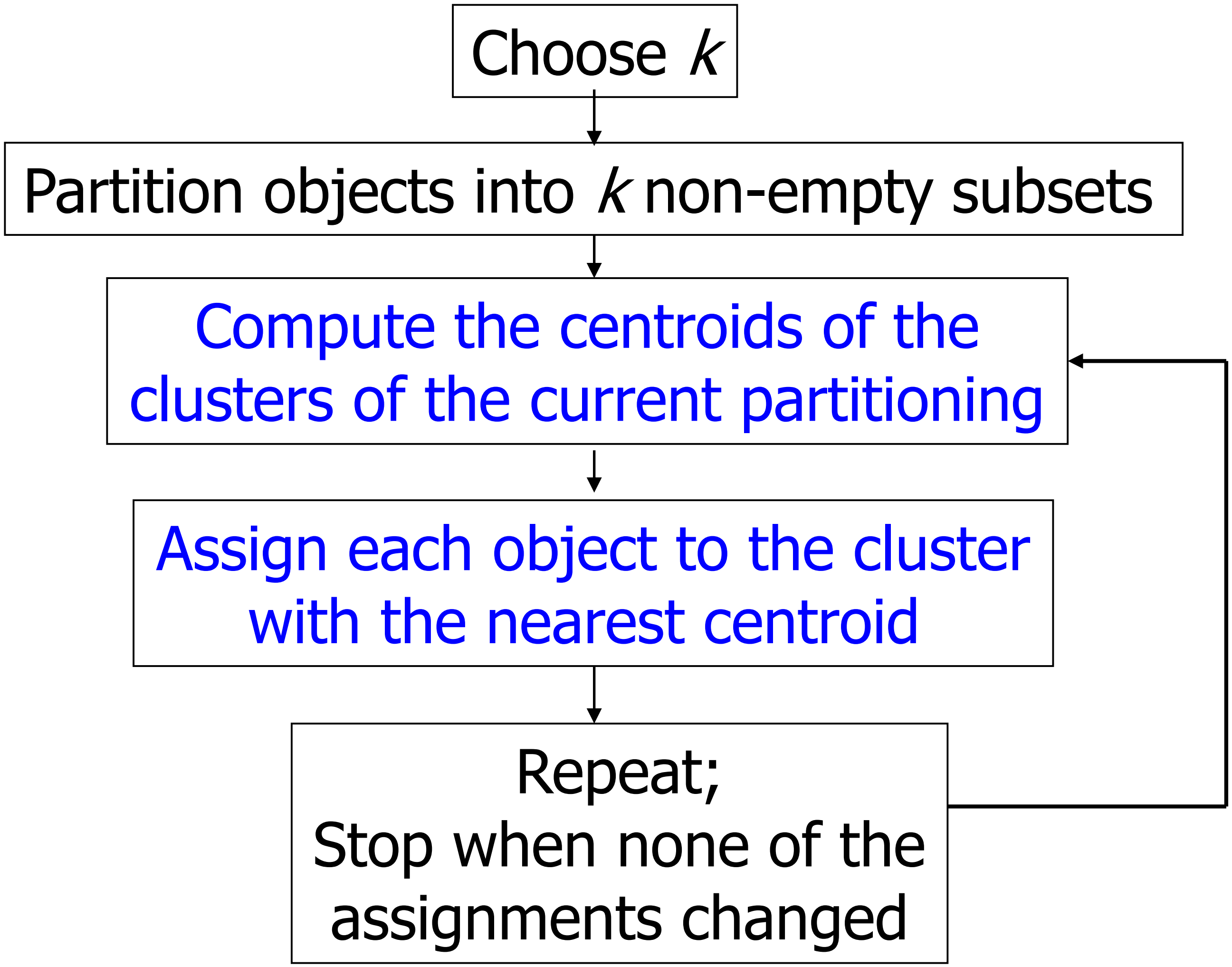
sum over clusters

each point *p* is assigned to exactly one cluster *i*

- Given *k*, find a partition of *k clusters* that **minimizes the error *E***
  - Find global optimum: exhaustively enumerate all possible partitions

    nearly impossible
  - Find a local optimum by heuristic methods:
    - ***k-means*** (MacQueen'67): Each cluster is represented by its center
    - ***k-medoids*** or Partition around medoids (PAM) (Kaufman&Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

k-means Clustering in Short



The *k*-means Algorithm

- Initialization
  - Set a value for k
  - Place the k cluster centres μ_j at random positions in input space
- Learning: Repeat …
  - For each data point x_p
    - Compute distance to each cluster centre
    - Assign data point to nearest cluster centre with distance

      $d_p = \min_j d(x_p, \mu_j)$
  - For each cluster centre
    - Move position of centre to mean of points in cluster

$$\mu_j = \frac{1}{N_j} \sum_{i=1}^{Nj} x_i \qquad N_j = \#\text{points in cluster } j$$

- … until the assignments don't change
  (then, cluster centres stop moving)

The *k*-means Algorithm – Online Version

- Initialization
  - Set a value for k
  - Place the k cluster centres μ_j at random positions in input space
- Learning: Repeat …
  - For each data point x_p
    - Compute distance to each cluster centre
    - Assign data point to nearest cluster centre with distance

      $d_p = \min_j d(x_p, \mu_j)$
    - Move position of centre slightly towards data point:

$$\mu_j \leftarrow \mu_j + \eta \cdot (x - \mu_j)$$

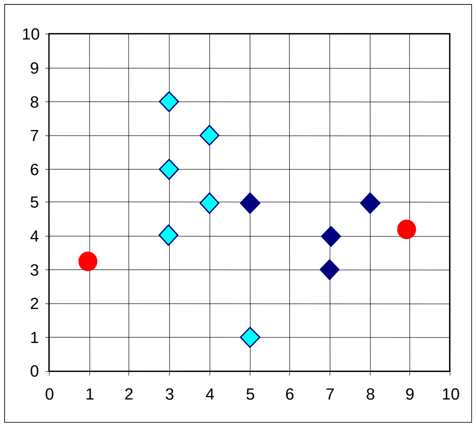
where η is a small learning rate.

Possibly: decrease $\eta = \eta(t)$ over time.

- … until assignments don't change … and a bit longer,
since clusters may keep moving slowly until converged

The *k-means* Clustering Method

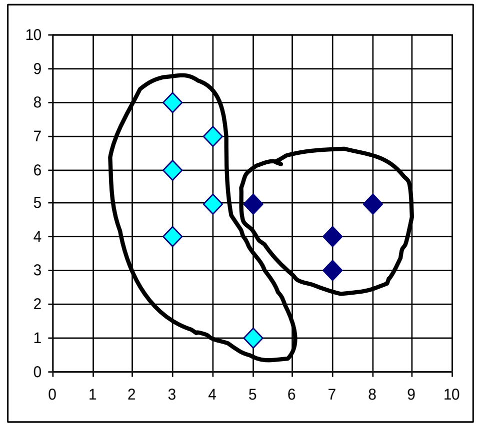
- Example



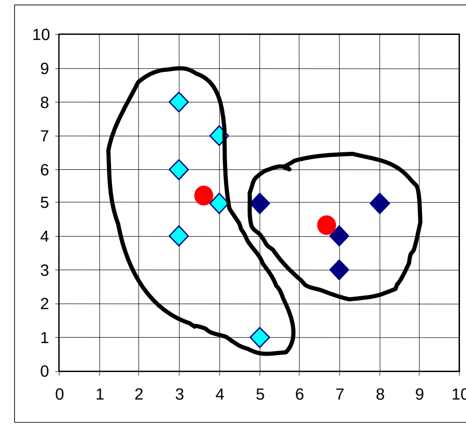
k=2

Arbitrarily choose *k* objects, or *k* locations, as initial cluster centers

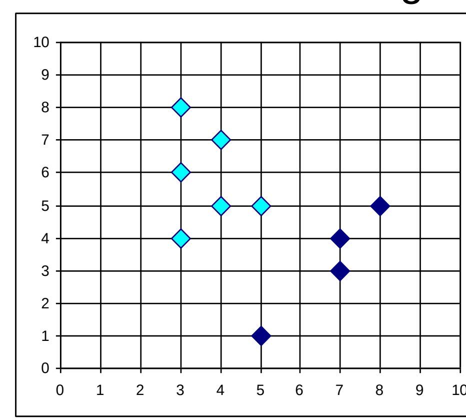
Assign each object to most similar center



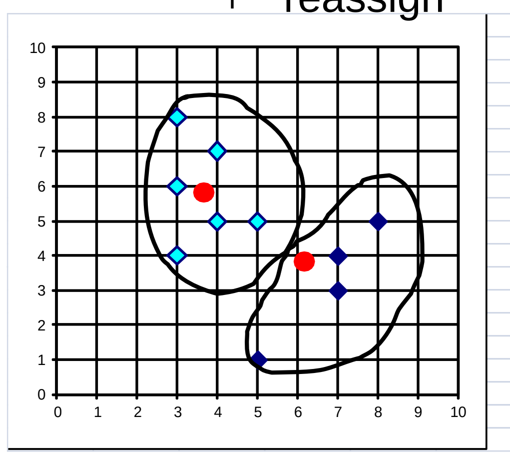
Update the cluster means



reassign



Update the cluster means

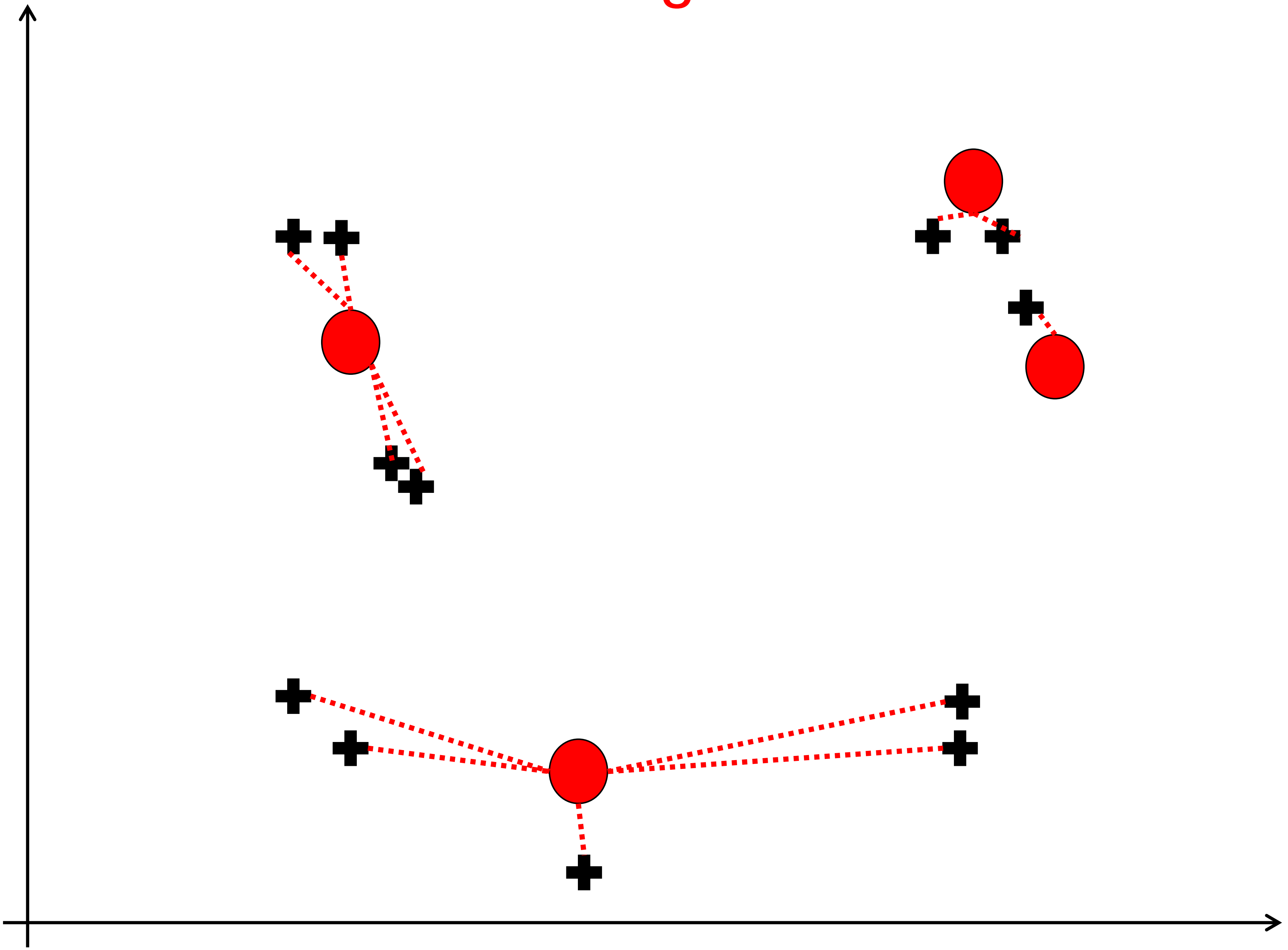


reassign

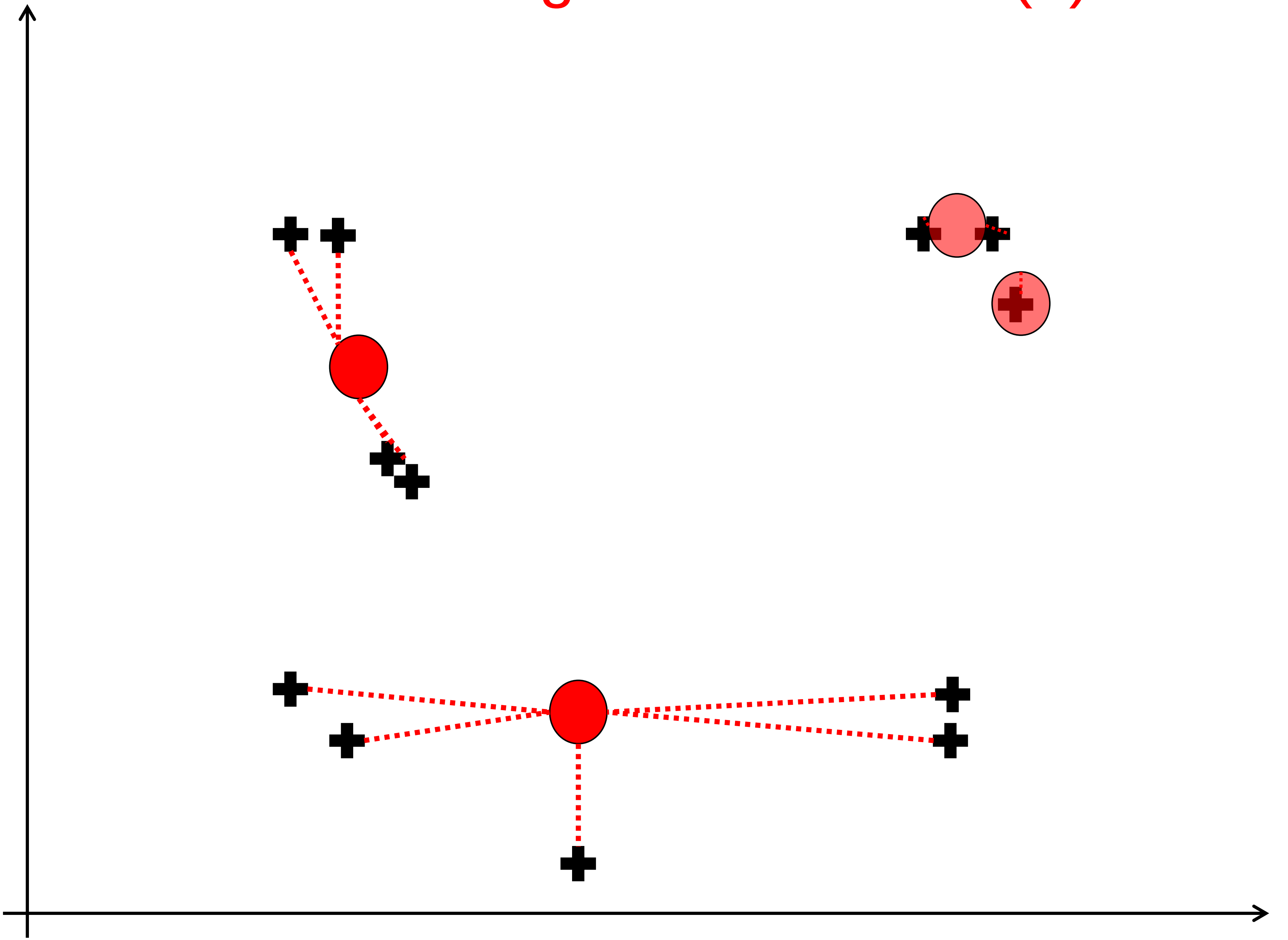
Why k-means Converges

- Change of assignments reduces the sum squared distances of the datapoints to their assigned cluster centers.
- Moving a cluster center reduces the sum squared distances of the datapoints to their assigned cluster centers.
- If the assignments do not change in the assignment step, we have converged.

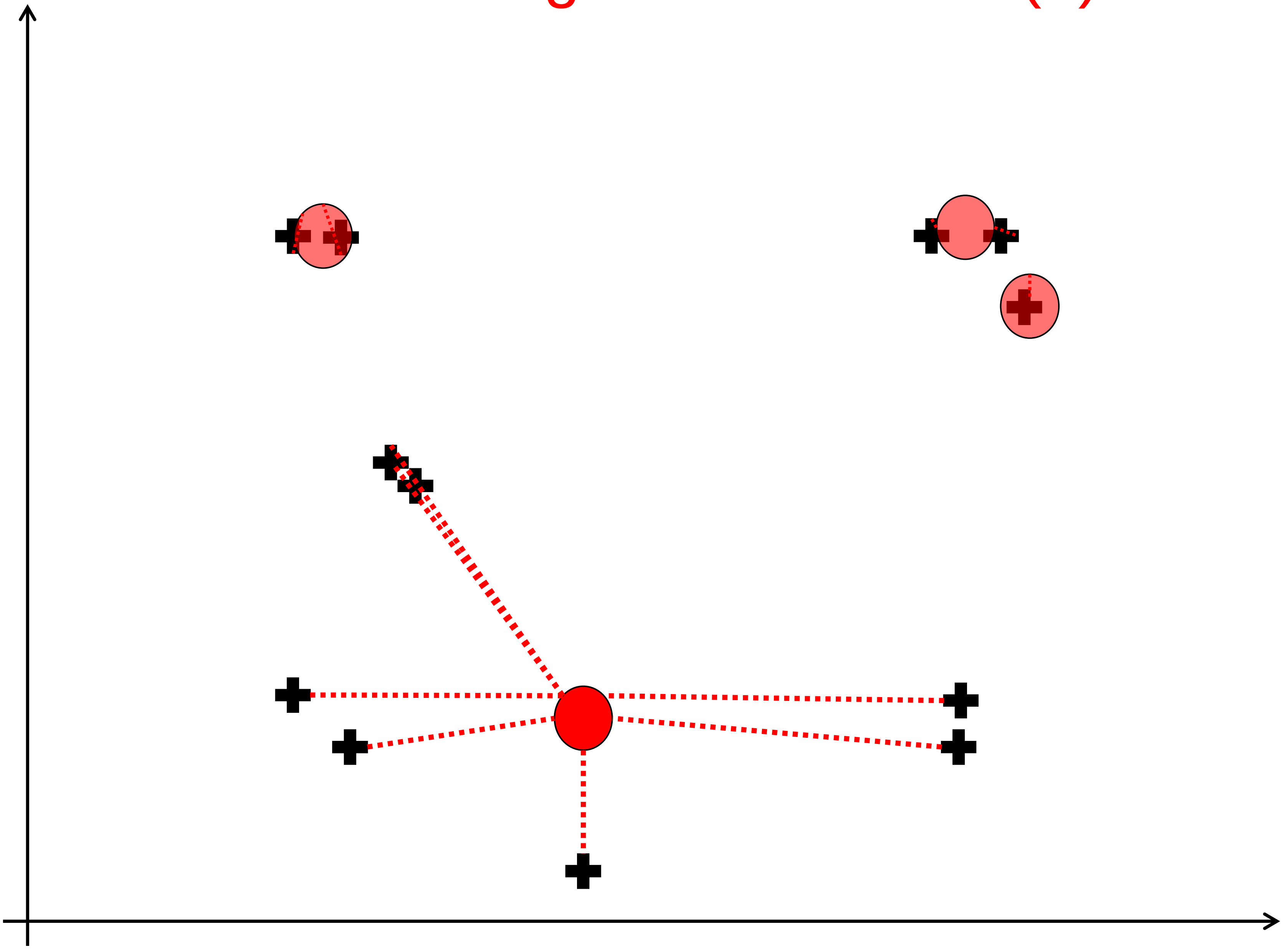
Clustering: 4-means



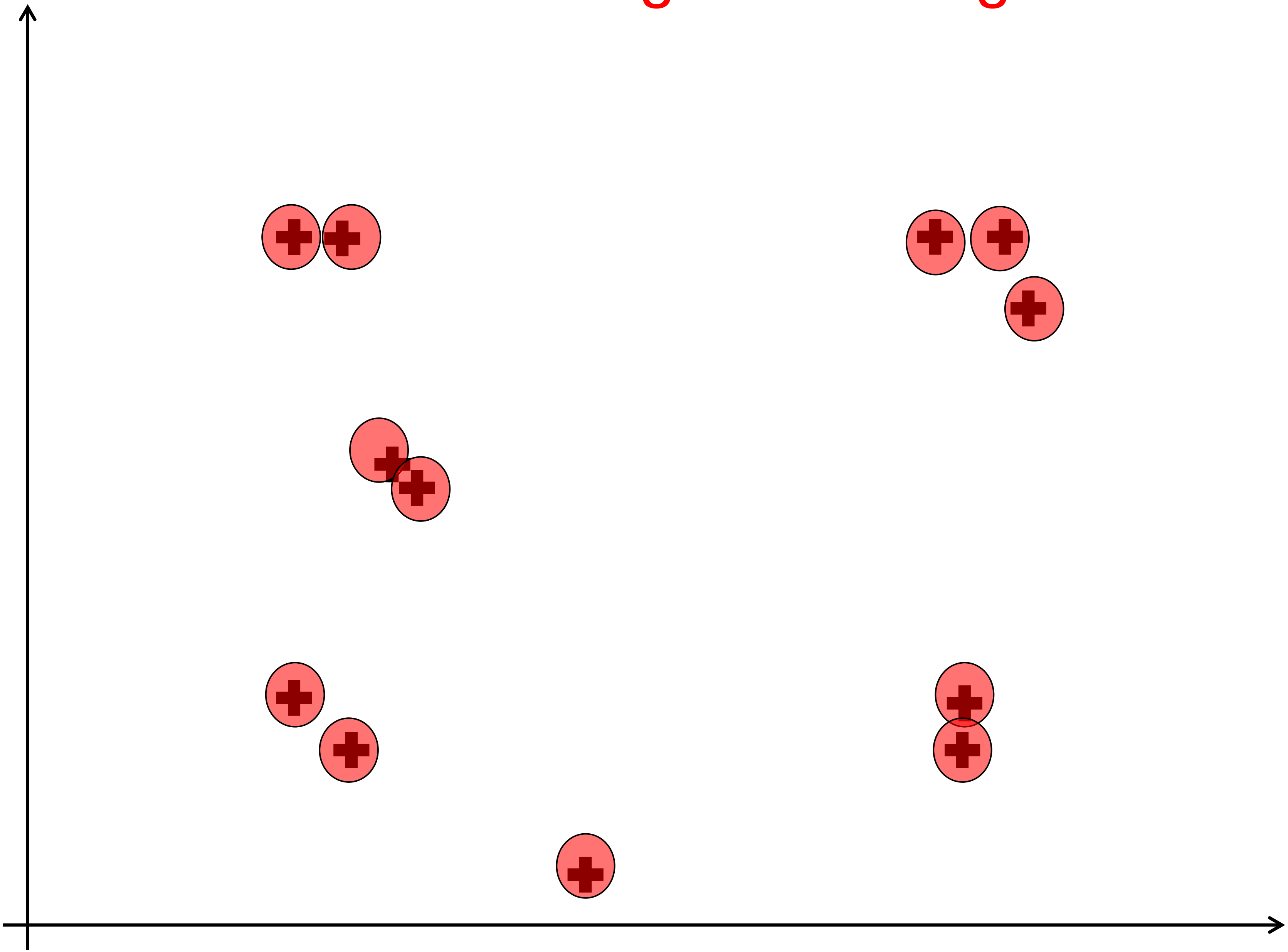
Clustering: Local Minima (1)



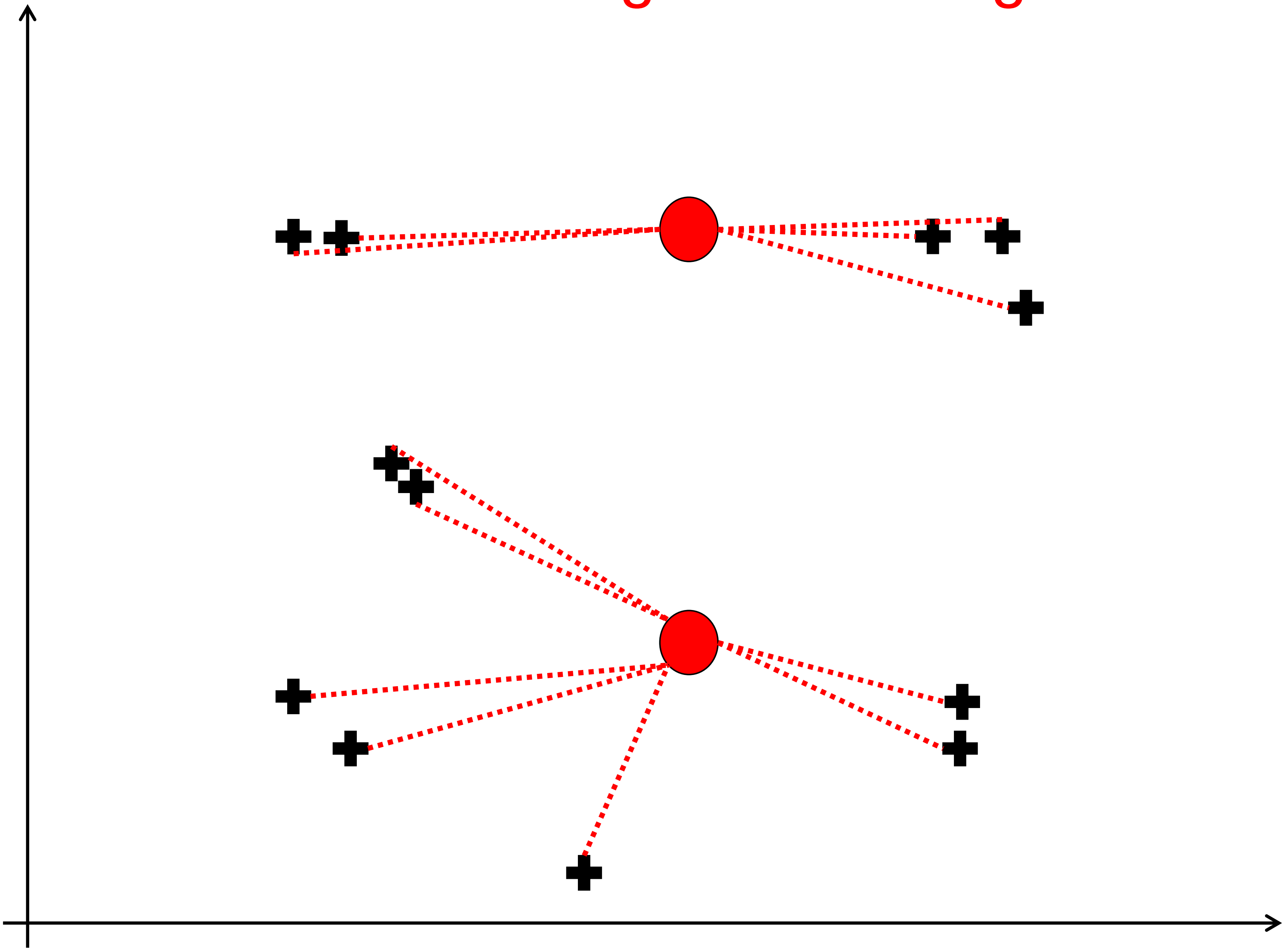
Clustering: Local Minima (2)



Clustering: Overfitting



Clustering: Underfitting



Comments on the *k-means* Method

- **Strength**: *Relatively efficient*: $O(tkn)$. Typically, k, t << n.
  (n = #objects, k = #clusters, t = #iterations)
- **Weaknesses**
  - Applicable only if *mean* is defined – not for categorical data
  - Not suitable to discover clusters with *non-convex shapes*
  - Sensitive to noisy data and *outliers*
  - Terminates at a ***local*** optimum

    → redo k-means with different initial cluster positions and choose the result that has minimal error E
  - Must set *k, number* of clusters, in advance

    → try different k for best cluster quality

    (e.g. Davies-Bouldin index)


THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA
ANSWERS

Example: Object Hypotheses in Natural Scenes using k-means

- In a stereo image pair of a scene, pixels can be clustered based on position, hue & saturation, and disparity.
- For object segmentation, if two objects are in close proximity, they are likely to be encapsulated by the same segment.
- If we give the information that a segment covers two objects, k-means (k=2) can find a likely split of that segment.
- Then the object modeling loop is resumed with the new hypotheses.

Object Hypotheses Example

Generating Object Hypotheses in Natural Scenes through Human-Robot Interaction

Niklas Bergström, Mårten Björkman, Danica Kragic

CSC/KTH Stockholm, Sweden

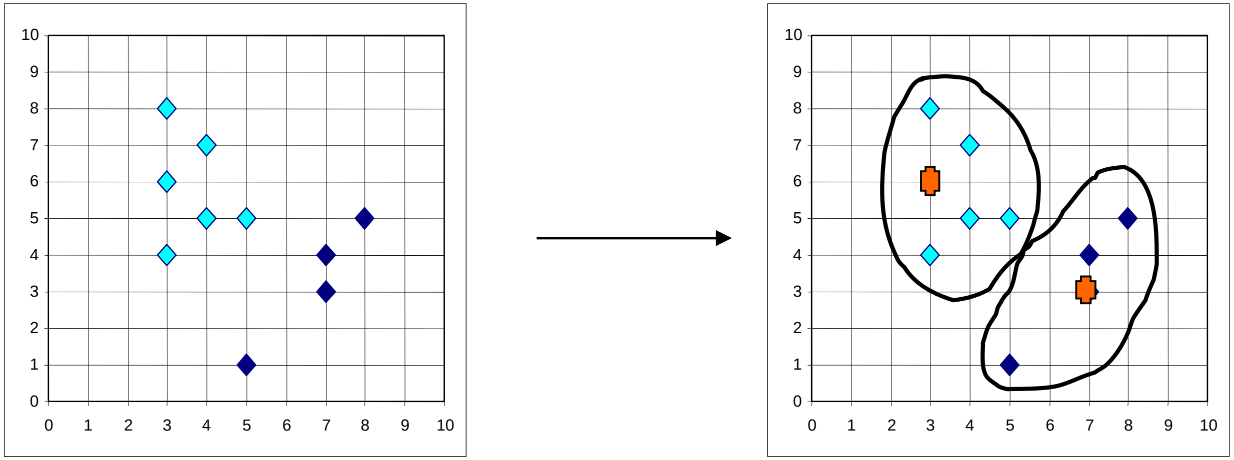
IROS '11

Clustering – Overview

- Background of clustering
  - Measure of cluster quality: Davies-Bouldin index
- K-means
- K-medoid
- Hierarchical clustering

Handling Outliers: the K-medoids Method

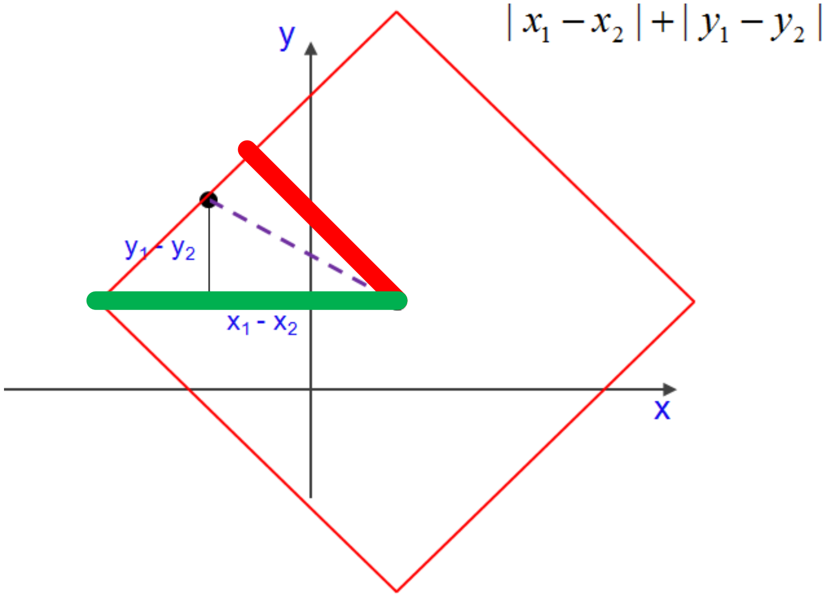
- ***K-medoids***: Instead of taking the *mean* value of the objects in a cluster as a reference point, *medoids* are used, which is the ***most centrally located object*** in a cluster.



- Variant: *K-medians* Clustering
  - For each cluster, use the *median in each dimension* of the data (the tuple of medians *may not correspond to a data object*)

Handling Outliers: the K-medoids Method

- ***K-medoids***: Instead of L_2 norm as in k-means (sensitive to outliers!), the L_1 norm, e.g. Manhatten distance is used
  - less sensitive to a larger difference in a single dimension
    - (in contrast, L_2 norm "amplifies" single-dimension large differences)
  - more sensitive to combined differences in multiple dimensions

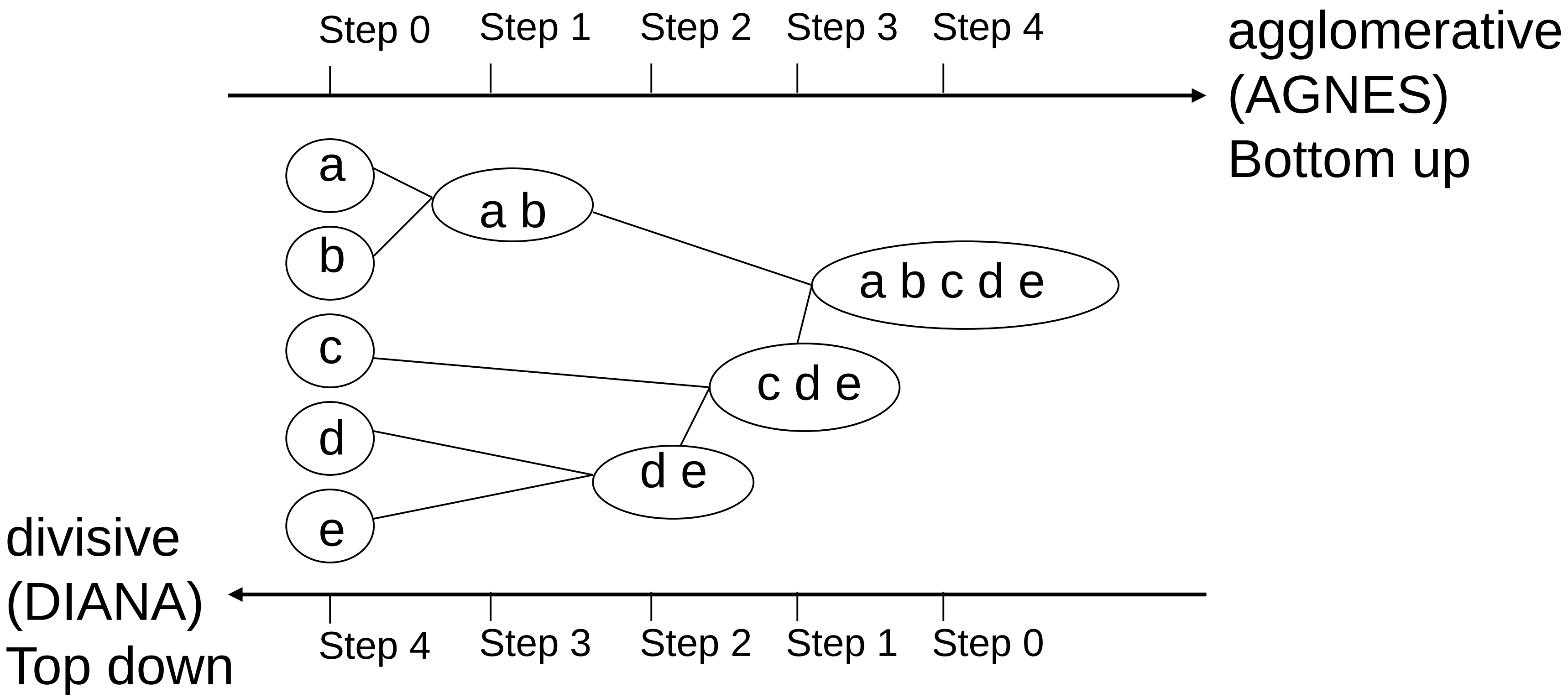


Clustering – Overview

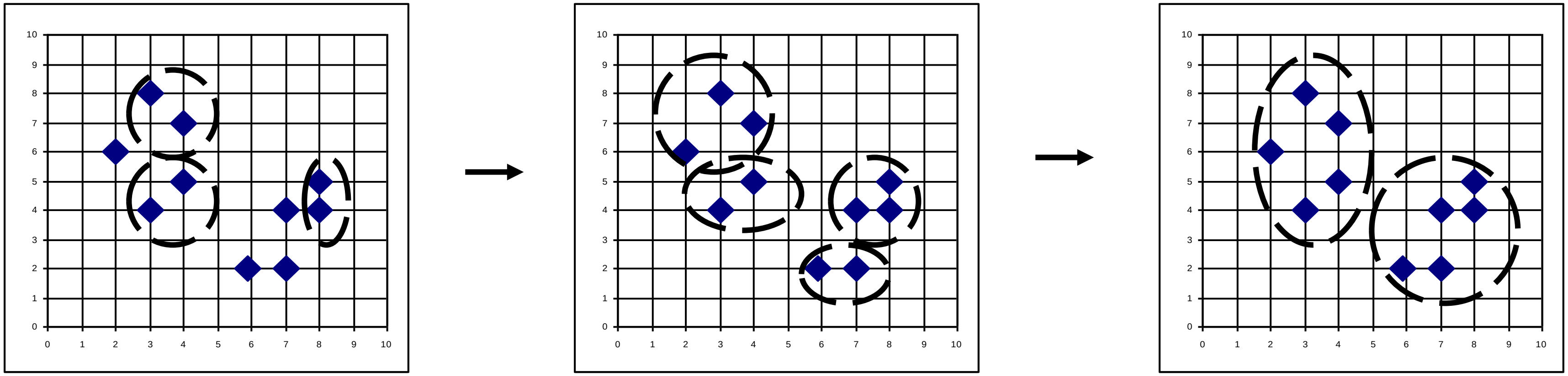
- Background of clustering
  - Measure of cluster quality: Davies-Bouldin index
- K-means
- K-medoid
- Hierarchical clustering

Hierarchical Clustering

- Use distance matrix as clustering criteria
- Does **not** require to set a fixed number *k* of clusters
- Instead, needs a termination condition



AGNES (Agglomerative Nesting)

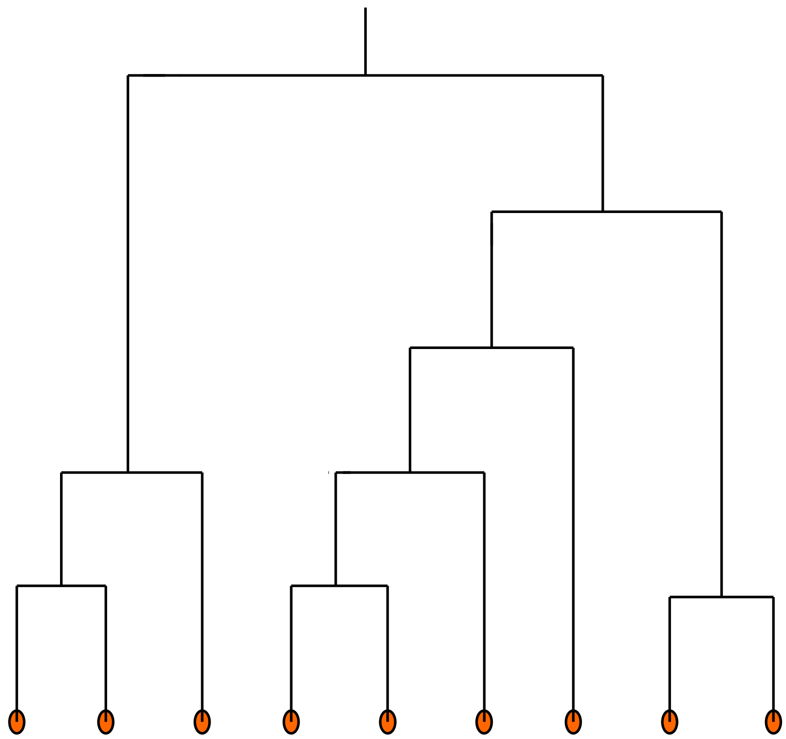


- Introduced in Kaufmann and Rousseeuw (1990)
- Implemented in statistical packages, e.g., Splus
- Use the ***single-link*** method and the dissimilarity matrix
- Merge nodes that have the least dissimilarity
- Go on in a non-descending fashion
- Eventually all nodes belong to the same cluster

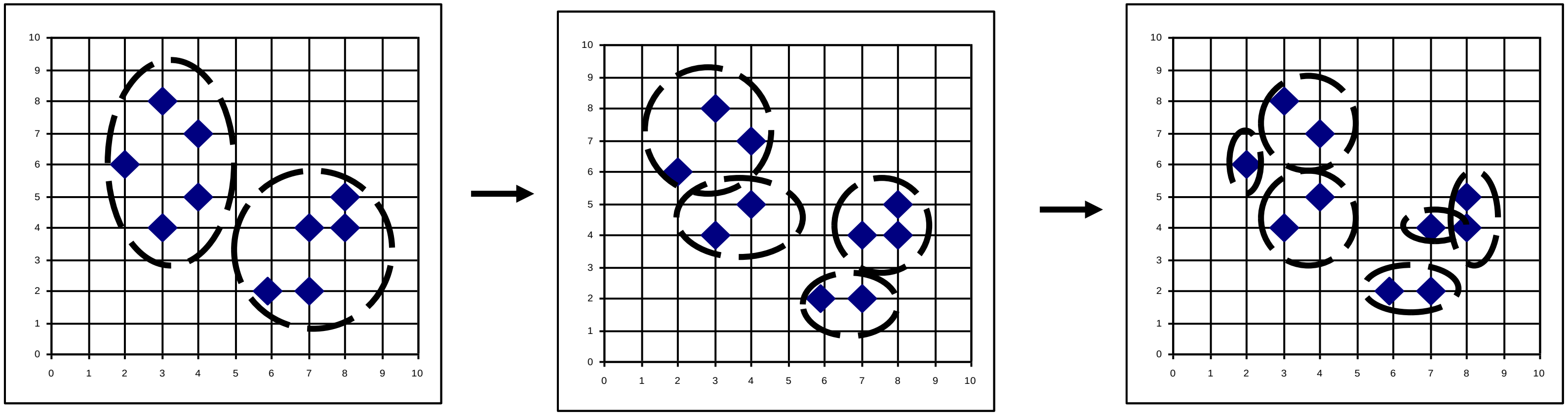
Dendrogram Shows how Clusters are Merged

- Decompose data objects into several levels of nested partitioning (***tree*** of clusters), called a ***dendrogram***.

- A ***clustering*** of the data objects is obtained by ***cutting*** the dendrogram at the desired level, then each ***connected component*** forms a cluster.



DIANA (Divisive Analysis)



- Introduced in Kaufmann and Rousseeuw (1990)
- Implemented in statistical analysis packages, e.g., Splus
- Inverse order of AGNES: top down
- Eventually each node forms a cluster on its own

Summary

- ***Cluster analysis*** groups objects based on their ***similarity*** while dissimilarity between clusters is also desired
- Similarity measures can be defined for ***various types of data***
- Wide applications, such as also

  → ***Outlier detection***, e.g. based on distance to cluster centre
- Clustering algorithms can be categorized into
  - partitioning, hierarchical methods (today)
  - neural network-based, density-based, grid-based, dimensionality reduction methods, …
- Still more research issues in cluster analysis