# Data Mining - Clustering

Nhat-Quang Doan

University of Science and Technology of Hanoi

*nq.doan@gmail.com*

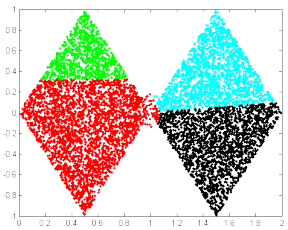
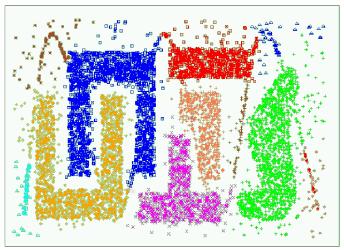
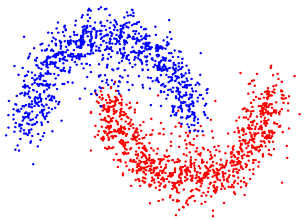
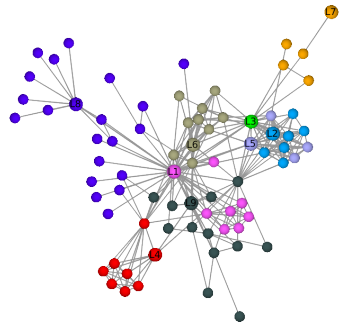
# Overview

# Introduction

## Objectives

- Clustering is the unsupervised learning task of data mining that retrieves information from data and determines the relationship between objects.
- Clustering consists in grouping a set of unlabeled data objects (instances) based on a similarity measure such that objects in the same *cluster* (group) are similar to each other and dissimilar to those in other clusters.

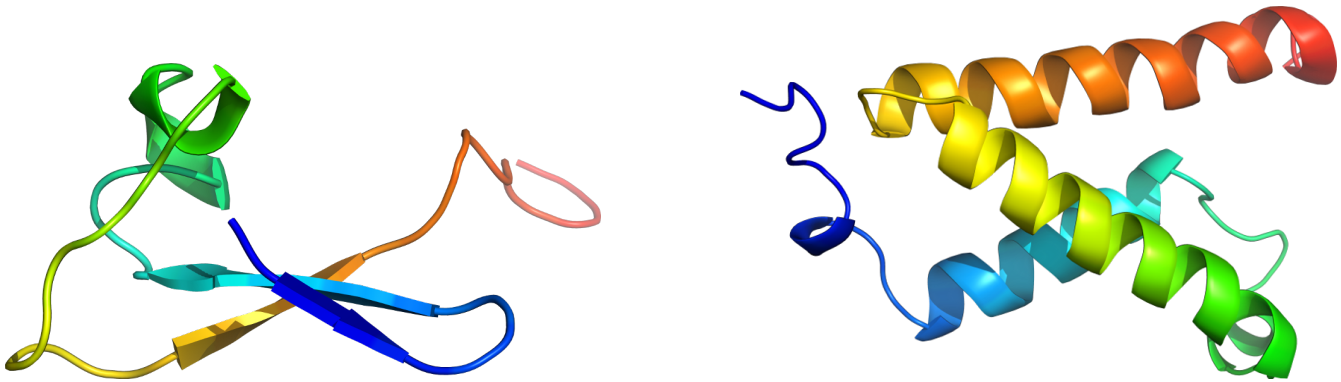
## Examples

# Introduction

## Applications

- Bio-informatics: detecting the cancer patients or finding abnormal genes

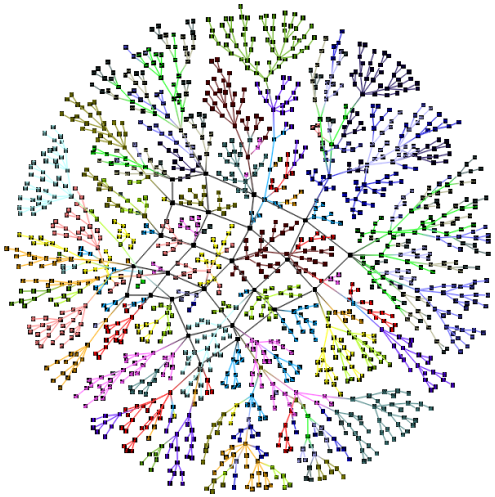


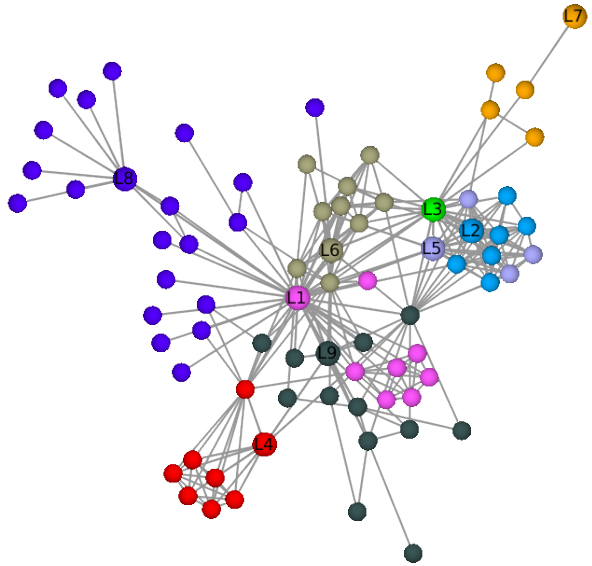
- Marketing: partitioning general customers' information.

# Introduction

## Applications

- Image Segmentation: locating the homogeneous zones in images.
- Text Mining: grouping texts, files, books.
- Social Network: detecting communities within large groups of people.





# Introduction

## Applications

- Content-Based Information Retrieval: Wang dataset

# Introduction

## Applications

- Image Segmentation

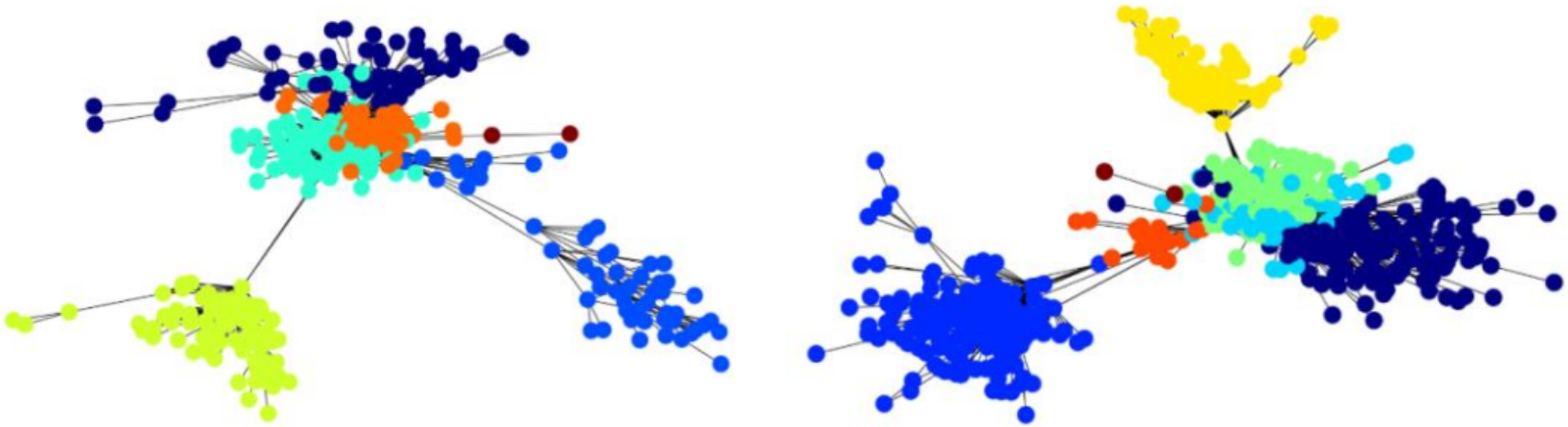
a

b

# Introduction

## Applications

- Recommendation System: Amazon dataset



a. k = 5 b. k = 6

**Fig. 2.** Visualization of clustering result for the Cell Phones and Accessories dataset.

# Problematics

## Difficulties

- Data nature: binary, graph, vector, tree, etc...
- Definition of similarity (or dissimilarity) measure between data objects.
- Clustering algorithms
- Big data
- Evaluation of a clustering result
- Data visualization

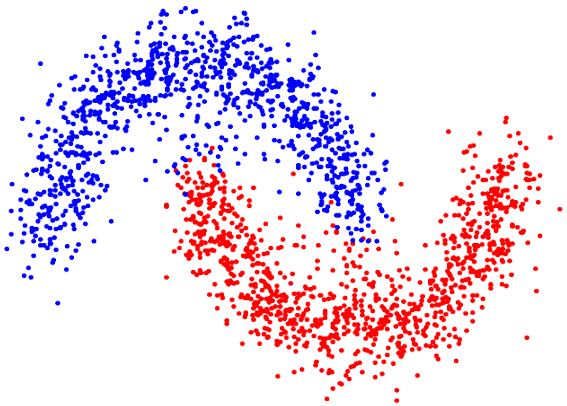
# Notions

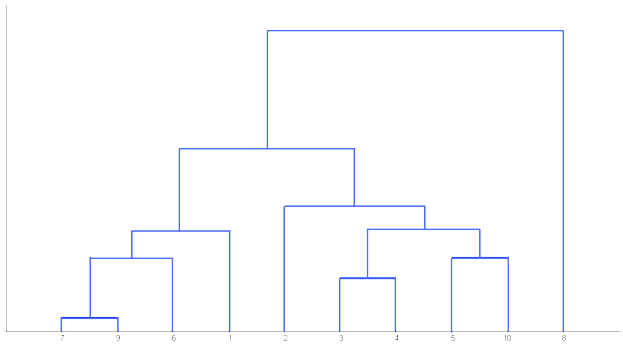
Input set: $\mathcal{X} = \{\mathbf{x}_i\}, \forall i = 1, .., n$: a set of $n$ objects
Object: $\mathbf{x}_i \in \mathbb{R}^d$ with $\mathbf{x}_i = (x_{i,1}, x_{i,2}, ..., x_{i,d})$
Cluster: $C_k \in \mathcal{C} = \{C_1, .., C_K\}, \forall k = 1, .., K$. The cardinality in each cluster $C_k$ is denoted by $n_k$.
Problem: Each object $\mathbf{x}_i \in \mathcal{X}$ is assigned to a cluster $C_k \in \mathcal{C}$ such that objects in $C_k$ are similar.

# Clustering algorithms

## Clustering algorithm categories:

- Centroid-based clustering algorithms: K-means, Self-organizing maps (SOM), Neural Gas.
- Hierarchical clustering algorithms: Agglomerative Hierarchical Clustering (AHC), BIRCH, CURE. Clustering tree

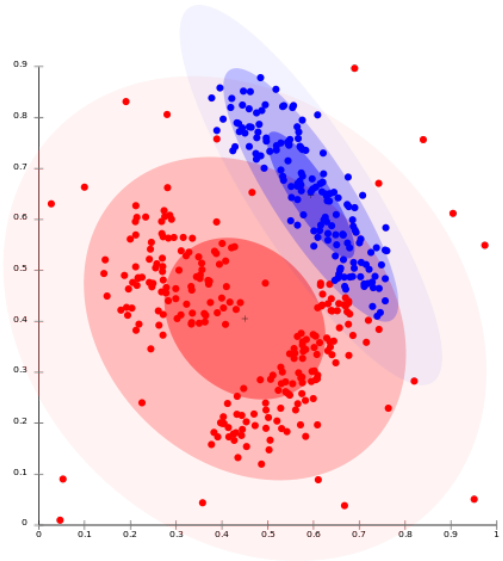


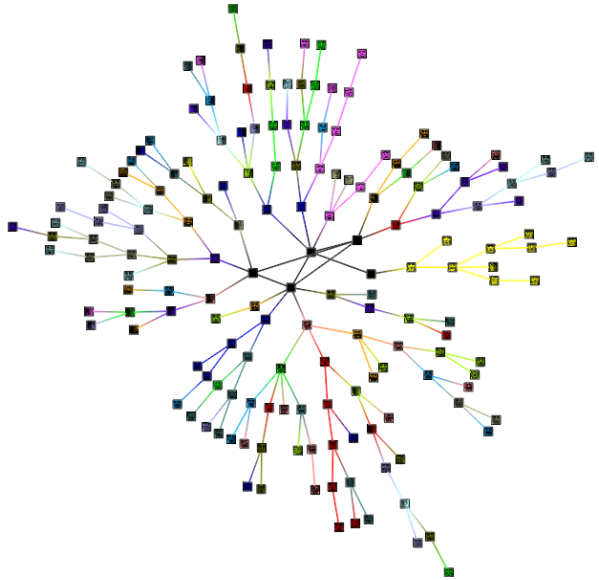


# Clustering algorithms

## Clustering algorithm categories:

- Density-based clustering algorithms: Expectation-Maximization.
- Hybrid clustering algorithms: AntTree, Self-organizing Trees (SoT).

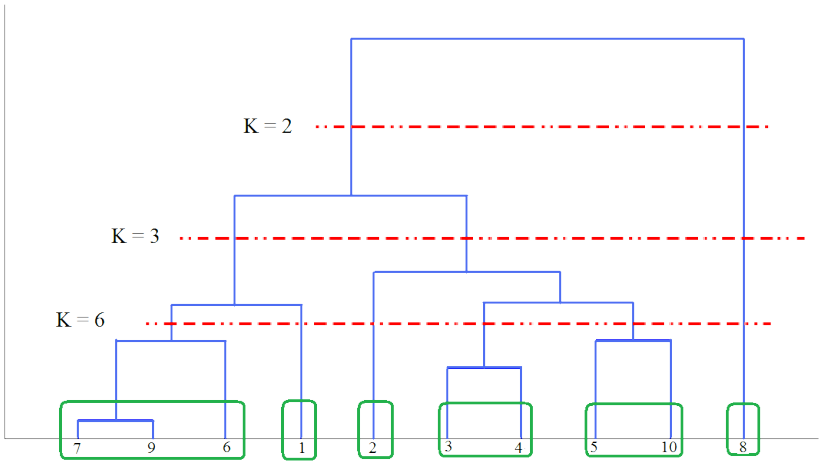


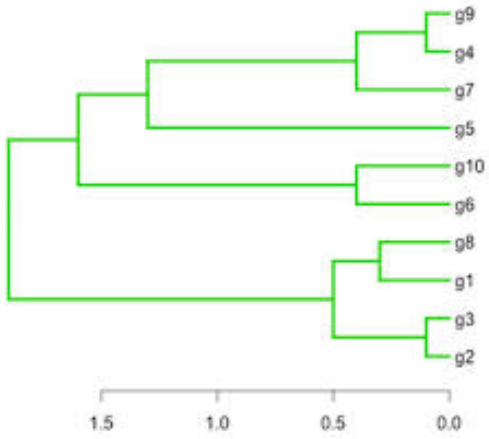


# Clustering algorithms: Hierarchical Clustering

The principles are as following:

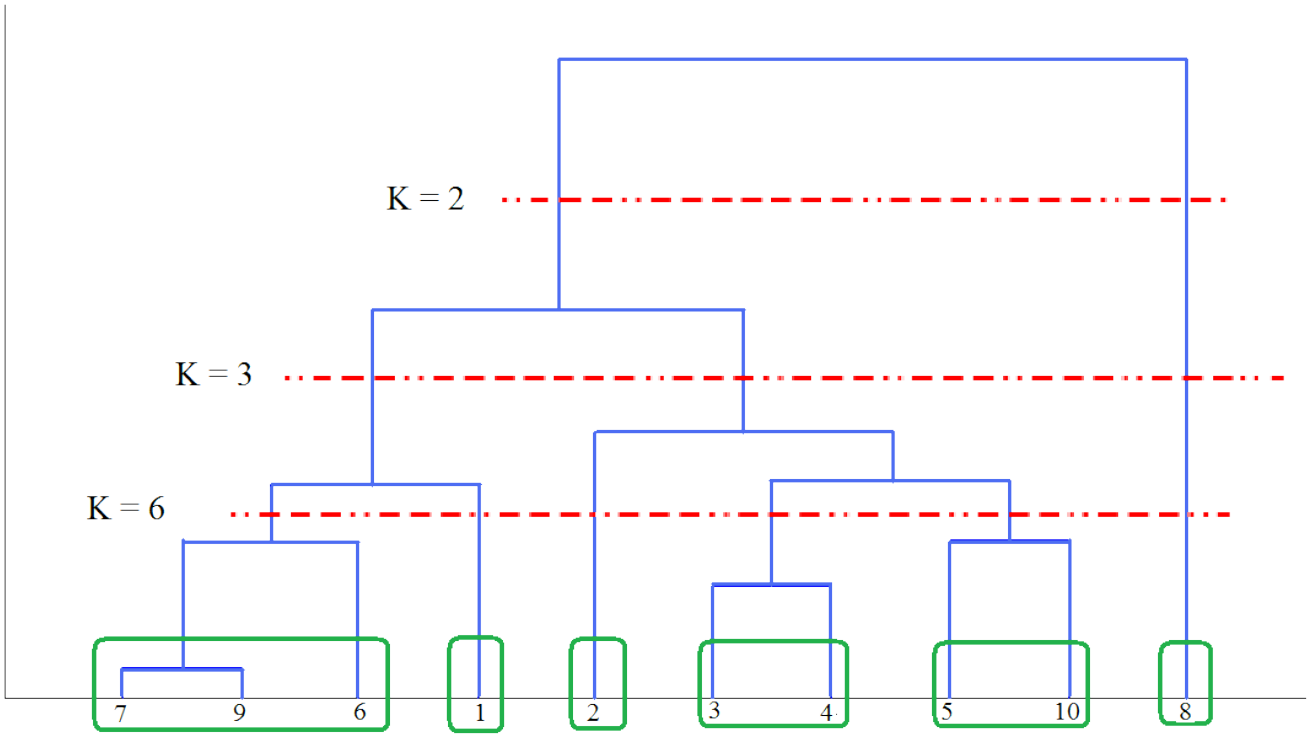
- A tree-based hierarchical taxonomy (dendrogram) is built from a set of objects.
- Each point or cluster is gradually "absorbed" by the nearest cluster.



# Clustering algorithms: Hierarchical Clustering

Clustering obtained by cutting the dendrogram at a desired level: each connected component forms a cluster.



# Clustering algorithms: Hierarchical Clustering

- Agglomerative (bottom-up):
  - Start with each object being a single cluster.
  - Eventually all objects belong to the same cluster.
- Divisive (top-down):
  - Start with all objects belong to the same cluster.
  - Eventually each object forms a cluster.
- Cluster height in the dendrogram corresponds to the similarity (distance) between two clusters before the merge.

# Clustering algorithms: AHC

**Algorithm 1** AHC algorithm

1: Each objects $\mathbf{x}_1, .., \mathbf{x}_N \in \mathcal{X}$ is in its own cluster $C_1, .., C_N$
2: **repeat**
3: merge the nearest clusters involving $C_i$ and $C_j$
4: **until** only one cluster is left

# Clustering algorithms: AHC

The nearest clusters can be variously defined as:

- Single-linkage: the link between two clusters is made by a single pair of objects that are closest to each other.

$$d(C_i, C_j) = \min_{\mathbf{x} \in C_i, \mathbf{x}' \in C_j} d(\mathbf{x}, \mathbf{x}')$$

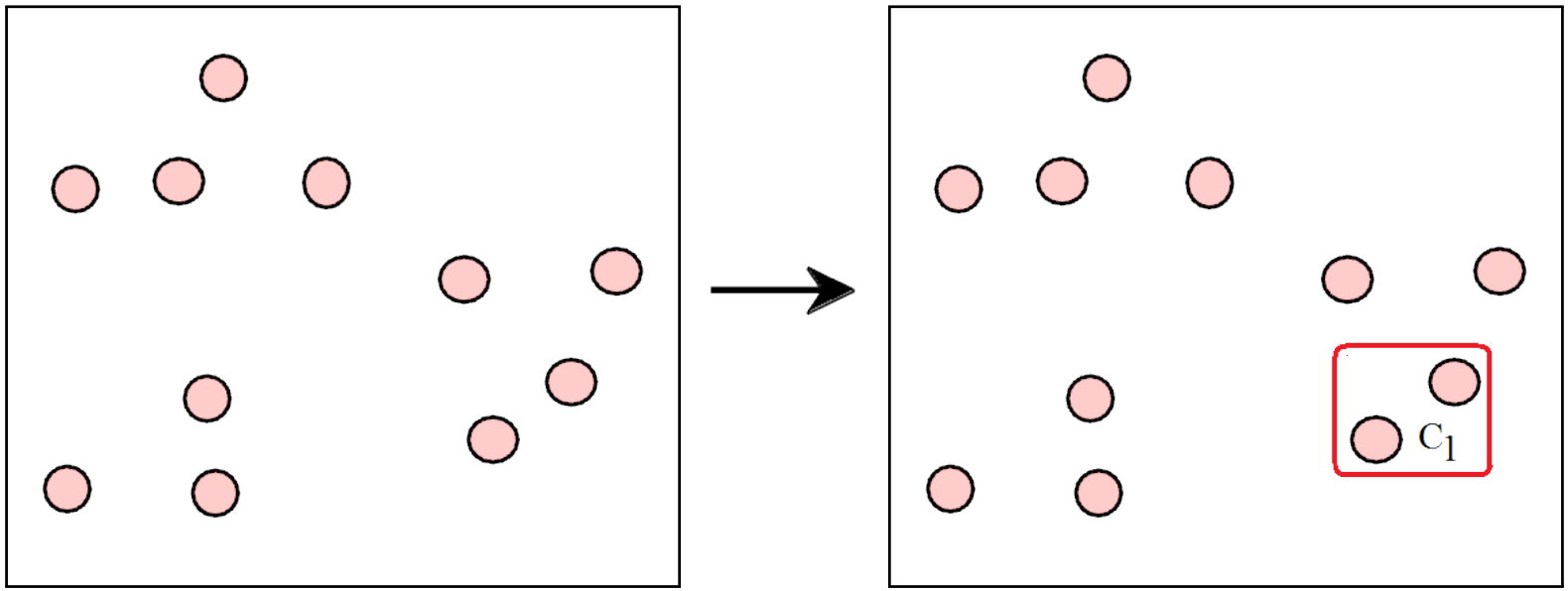
- Complete-linkage: the distance between clusters is equal to the distance between those two objects that are farthest away from each other.

$$d(C_i, C_j) = \max_{\mathbf{x} \in C_i, \mathbf{x}' \in C_j} d(\mathbf{x}, \mathbf{x}')$$

- And the other distances can be employed.

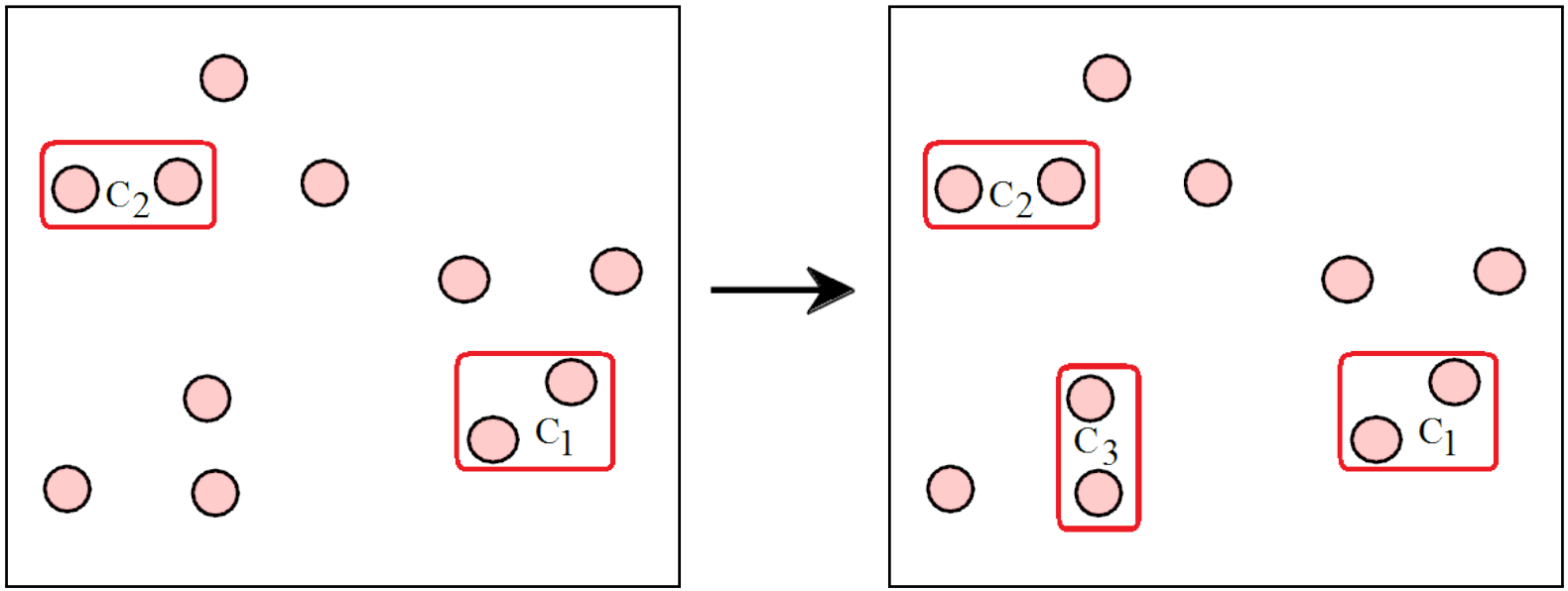
# Clustering algorithms: AHC

## Example



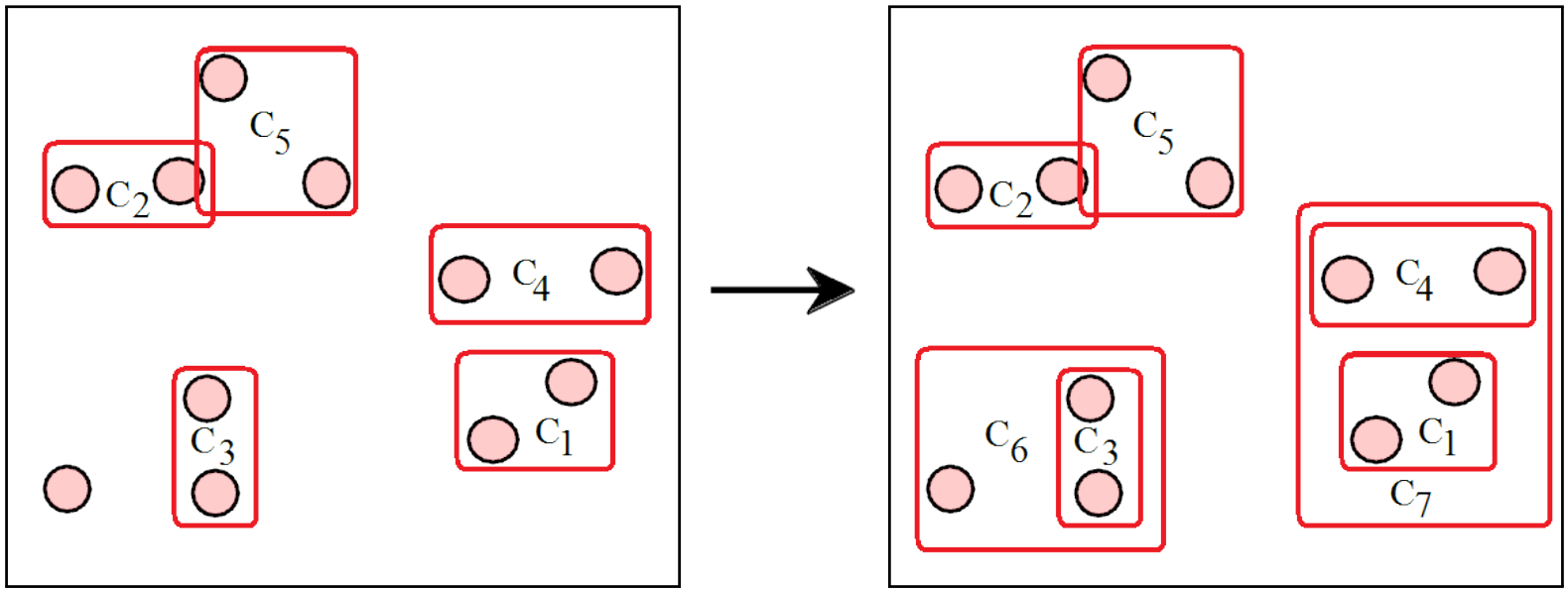
# Clustering algorithms: AHC

## Example



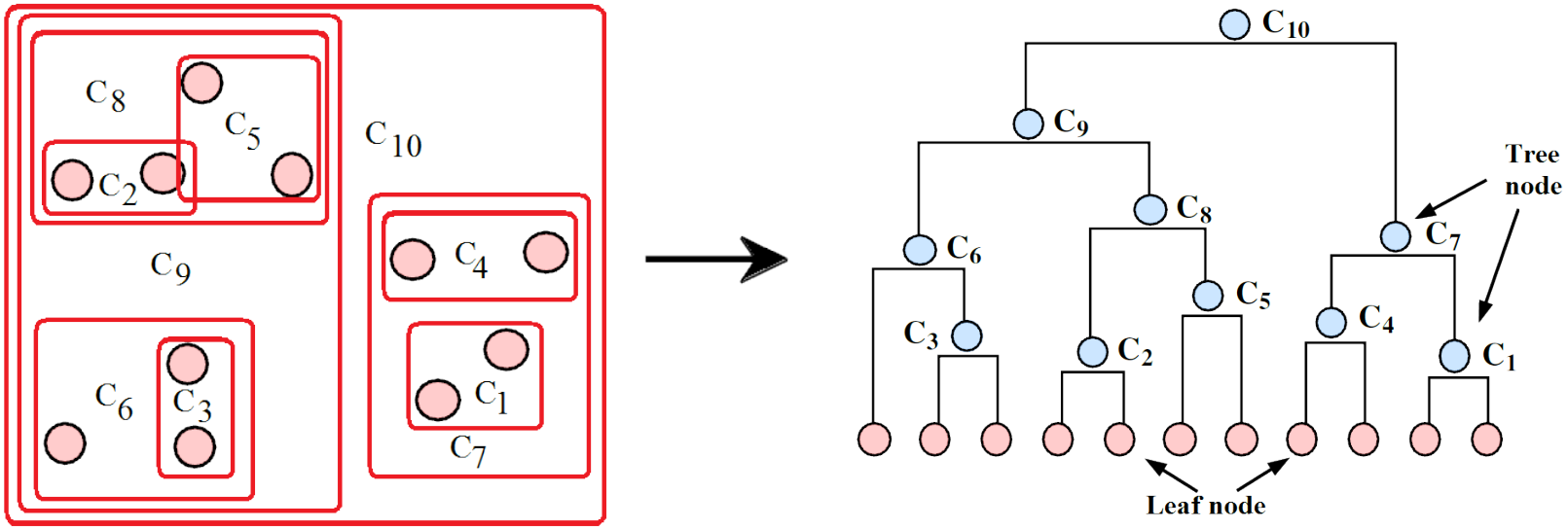
# Clustering algorithms: AHC

## Example



# Clustering algorithms: AHC

## Example



# Clustering algorithms: AHC

## Advantages

- does not require any input parameters in advance.
- simple visualization and easy comparison in similarity between objects.

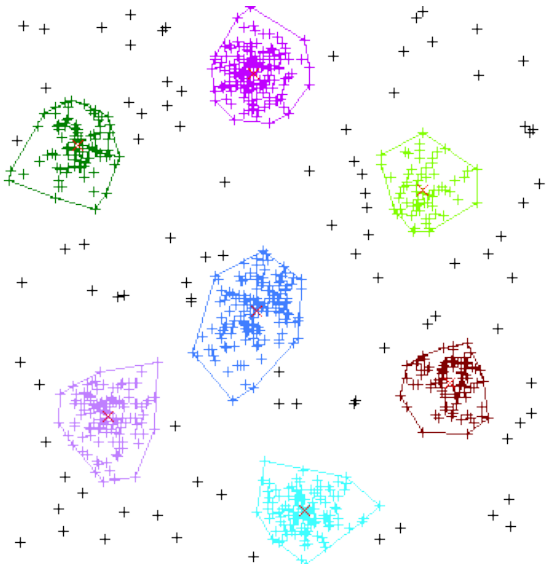
## Drawbacks

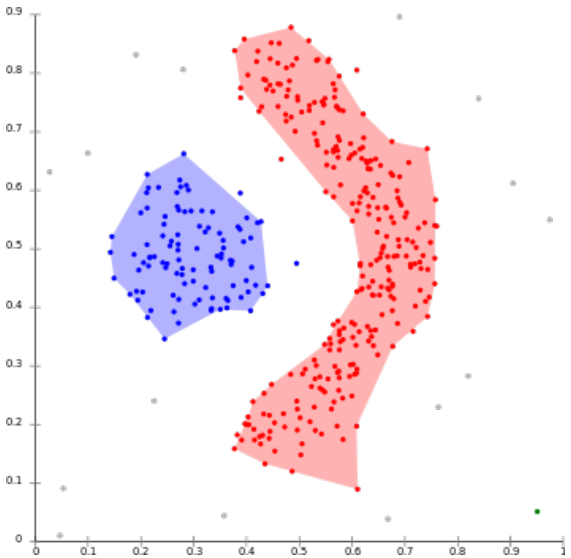
- high complexity $O(n^2)$ or $O(n^3)$.
- sensible to noisy.
- does not scale well, check all the number of data before splitting.

# Clustering algorithms: DBSCAN

Density-Based Spatial Clustering of Application with Noise groups points that are closely packed together (many nearby neighbors).

- If an object $\mathbf{x}_i$ is **density connect** to $\mathbf{x}_j$, then $\mathbf{x}_i$ and $\mathbf{x}_j$ belong to the same cluster.
- If an object $\mathbf{x}_i$ is **not density connect** to any other object, $\mathbf{x}_i$ is considered as noise.

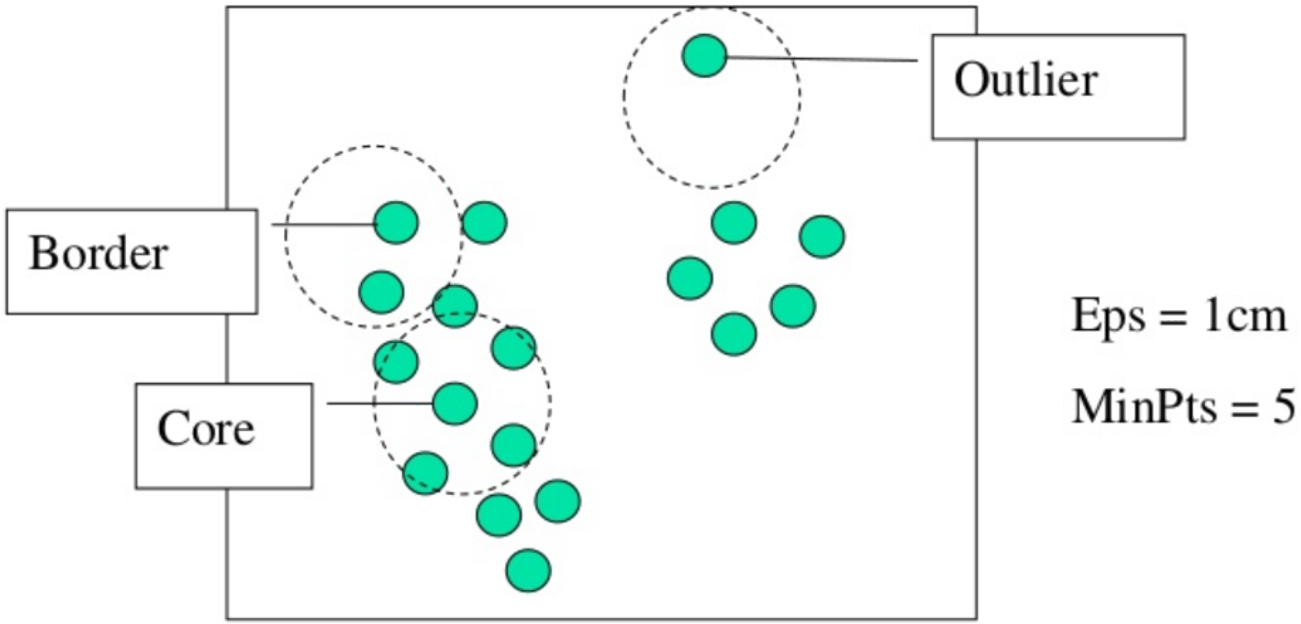




# Clustering algorithms: DBSCAN

- $\epsilon$-neighborhood: The $\epsilon$-neighborhood of an object p is the set of object within $\epsilon$-distance of $\mathbf{x}_i$
- core object: An object $\mathbf{x}_i$ is a core object if and only if there are at least minPts objects within the radius of $\epsilon$.
- noise object: An object $\mathbf{x}_i$ is not core object then it is a noise object.

# Clustering algorithms: DBSCAN




Sample for different defined objects in DBSCAN

# Clustering algorithms: DBSCAN

1. Classify objects as noise or core
2. Eliminate noise objects
3. Perform clustering on the core objects

```
current_Cluster_Label ← 1
for all core objects do
    if the core object x_i has no cluster label then
        current_Cluster_Label ← current_Cluster_Label + 1
    end if
    for all objects x_j in the radius of ε, except x_i do
        if the core object x_j has no cluster label then
            Label the object with cluster label current_Cluster_Label
        end if
    end for
end for
```

# Clustering algorithms: $K$-means

The principles are as following:

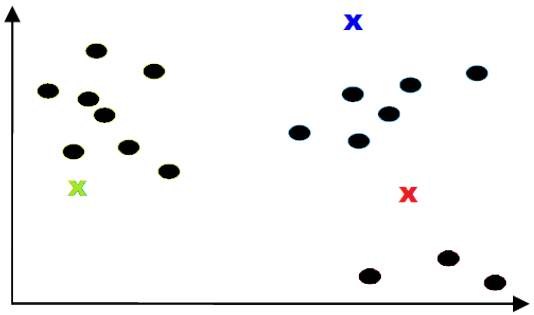
- Given that $K$ centroids $\mathcal{W} = \{\mathbf{w}_1, .., \mathbf{w}_K\}$ are known, an object $\mathbf{x}_i, \forall i = 1, .., n$ is assigned to the nearest centroid to minimize the quantization error.
- Once all the input objects have been assigned, for each cluster $C_k, \forall k = 1, .., K$, we estimate the new centroid $\mathbf{w}_k$.
- These steps are repeated until the centroids are convergent.

# Clustering algorithms: $K$-means

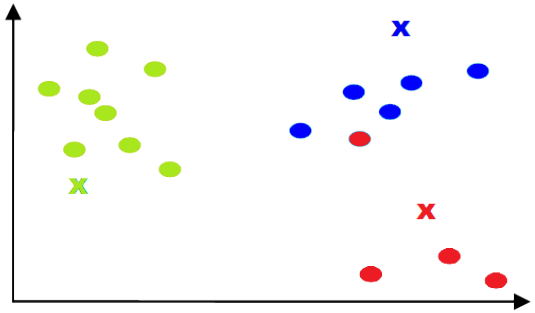
**Algorithm 2** K-means algorithm

initialize randomly $K$ prototypes / weight vectors
**repeat**
  **for** $i = 1$ **to** $n$ **do**
    $k = \arg\min_{k=1,..,K} \|\mathbf{x}_i - \mathbf{w}_k\|^2$
    $C_k = C_k \cup \mathbf{x}_i$ {assign $\mathbf{x}_i$ to cluster $C_k$}
  **end for**
  **for** $k = 1$ **to** $K$ **do**
    $\mathbf{w}_k = \frac{1}{n_k} \sum_{j=1}^{n_{C_k}} \mathbf{x}_j$ {update prototype $k$, where $n_k$ is the cardinality of cluster $C_k$}
  **end for**
**until** stopping criterion has been fulfilled

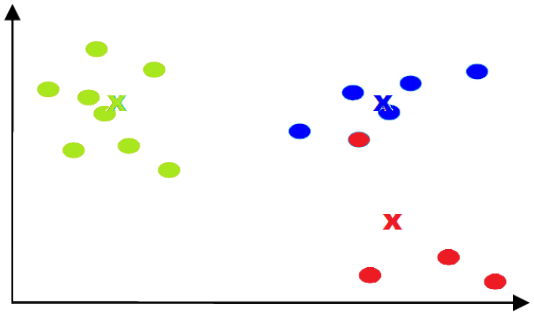
# Clustering algorithms: $K$-means



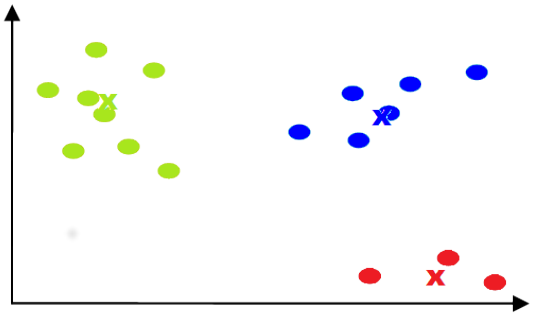
(a) Initial step



(b) Assignment step

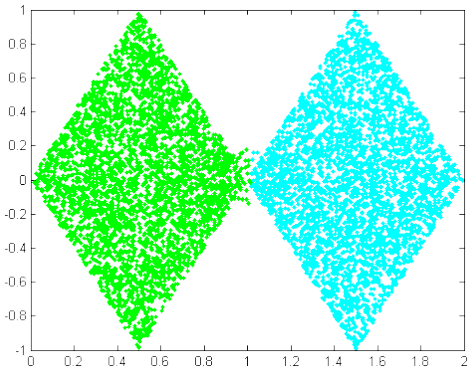


(c) Update step

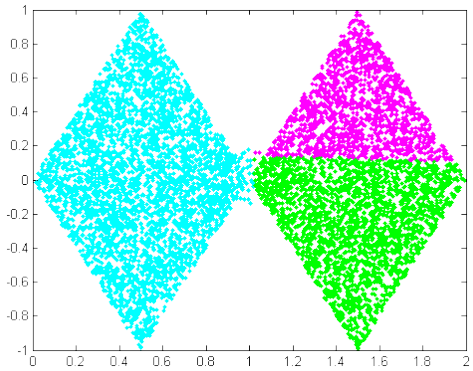


(d) Convergence.

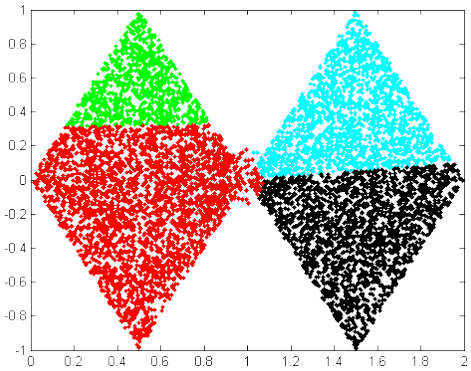
# Clustering algorithms: $K$-means



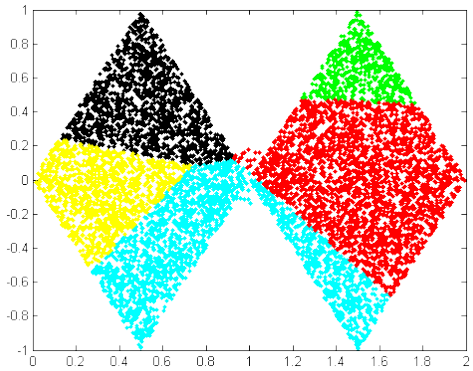
(a) $K = 2$



(b) $K = 3$



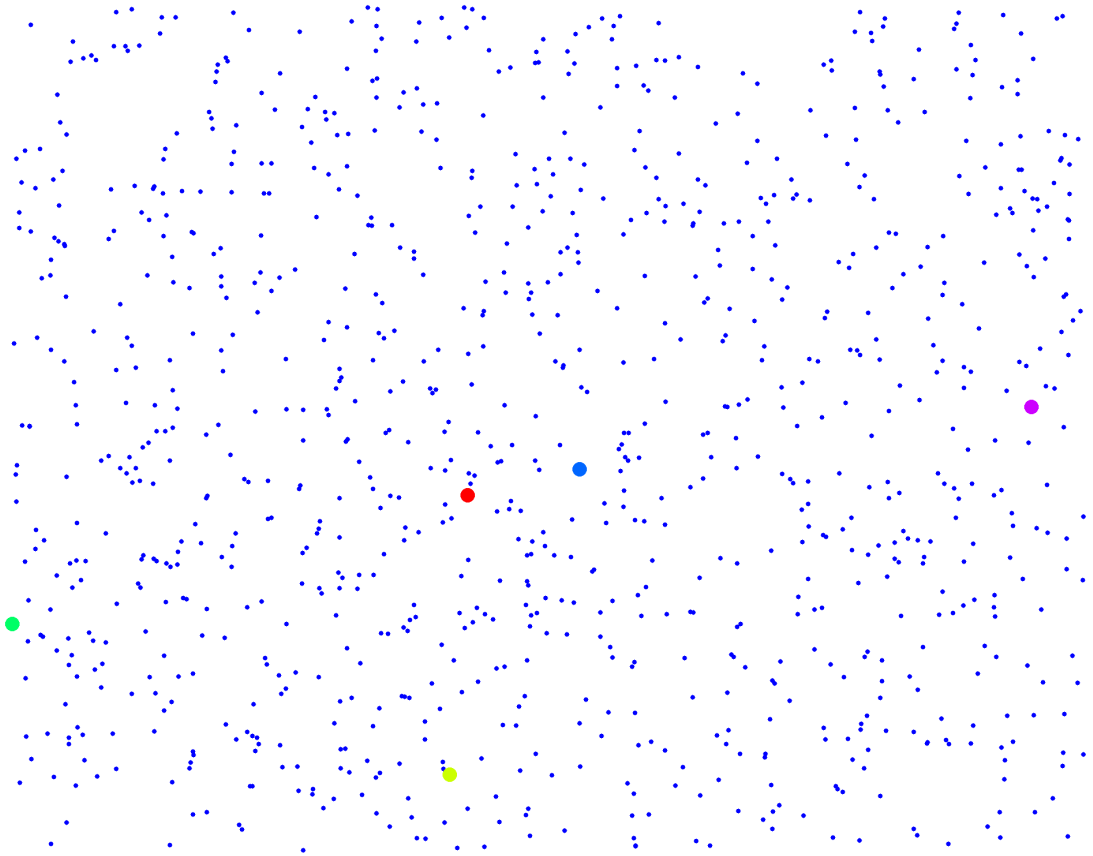
(c) $K = 4$



(d) $K = 5$

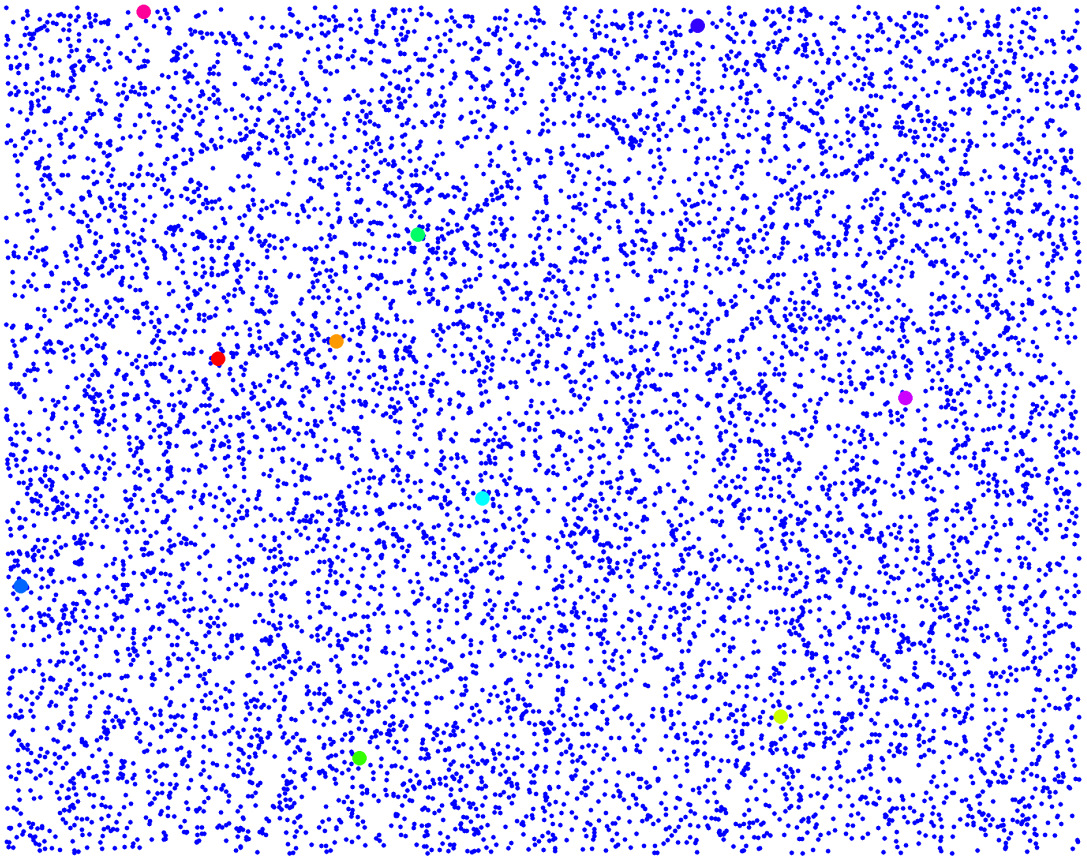
# Clustering algorithms: $K$-means

1000 data and 5 clusters



# Clustering algorithms: $K$-means

10000 data and 10 clusters



# Clustering algorithms: $K$-means

## Advantages:

- Simple algorithm and easy implementation.
- Low complexity $\theta(Knt)$ ($t$ is the number of iterations).

## Drawbacks:

- This algorithm depends greatly on the initialization:
  - $K$ must be defined a priori. Which value is the best for a specific data set?
  - $\mathbf{w}_k$ is randomly generated in the input space, it converges to the local minimum. How to initialize centroids to obtain better result?

# Clustering algorithms: $K$-means

To obtain a *stable* clustering result:

- Vary the values of $K$.
- Run the algorithm many times with random initialization of prototypes.
  $\rightarrow$ Group all the objects that are found together in the same cluster.

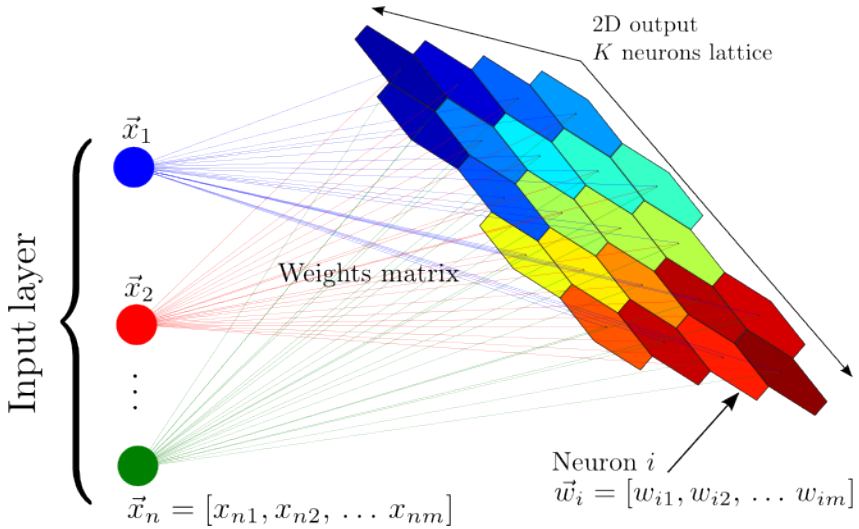
$K$-means variants:

- K-medians
- Self-organizing map
- Neural Network
- etc...

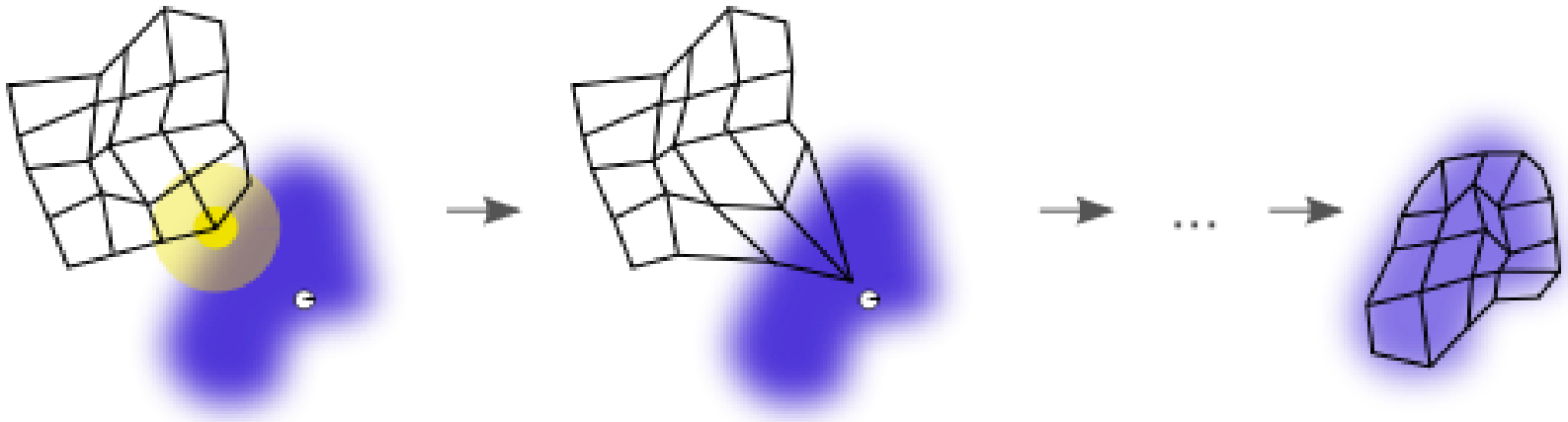
# Clustering algorithms: Self-organizing Map

## Introduction

- Introduce by Prof. Teuvo Kohonen in 1982
- Unsupervised neural network based on K-means
- Visualization tool for high-dimensional and complex data



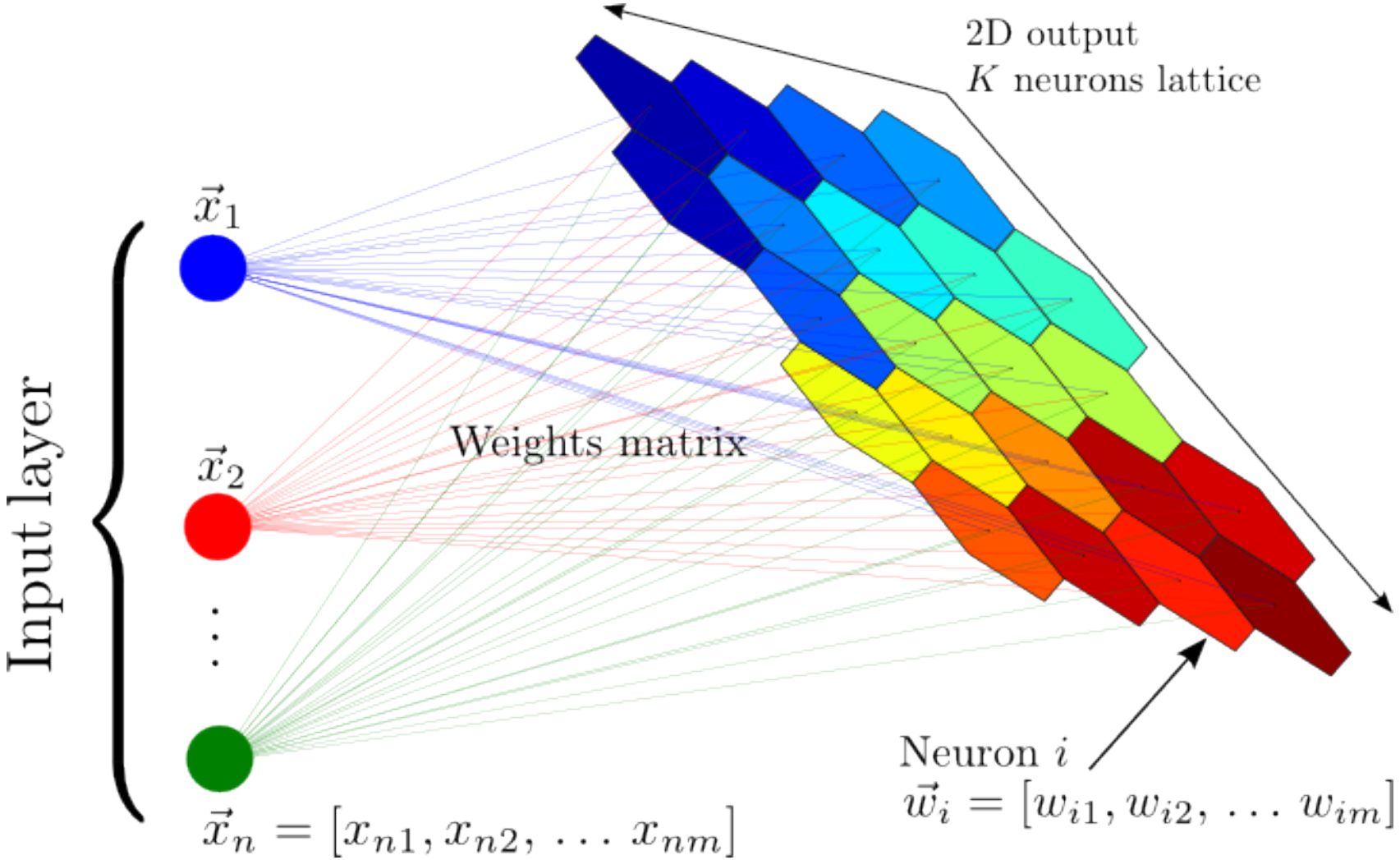
# Clustering algorithms: Self-organizing Map



The idea:

- Cover the input data distribution by a topology map
- Similar data in a cluster is represented by map nodes (neurons)

# Clustering algorithms: Self-organizing Map



# Clustering algorithms: Self-organizing Map

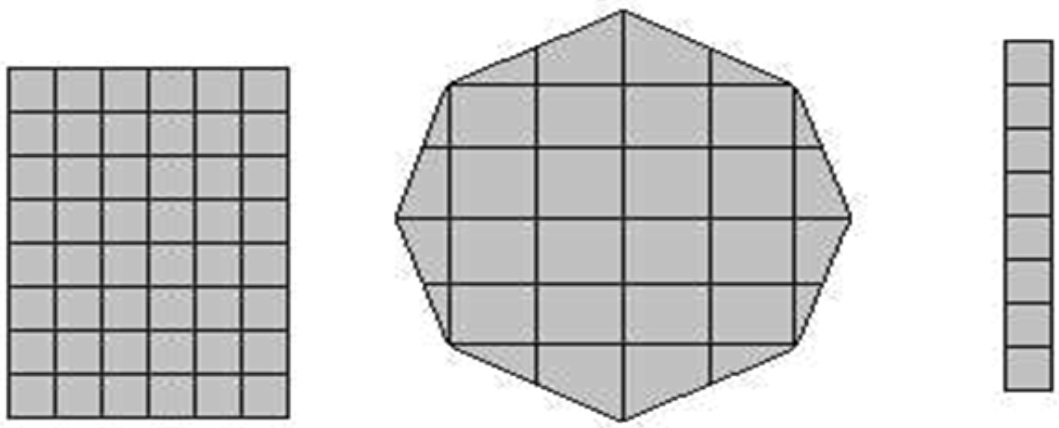
Four steps like K-means:

- Initialization: define the map topology, neurons etc...
- Assignment: assign data to the best match units (prototype)
- Update: recompute the new prototypes
- Convergence: repeat the above steps until the stopping criteria have been fulfilled

# Clustering algorithms: Self-organizing Map

SOM initialization: definition of map structure

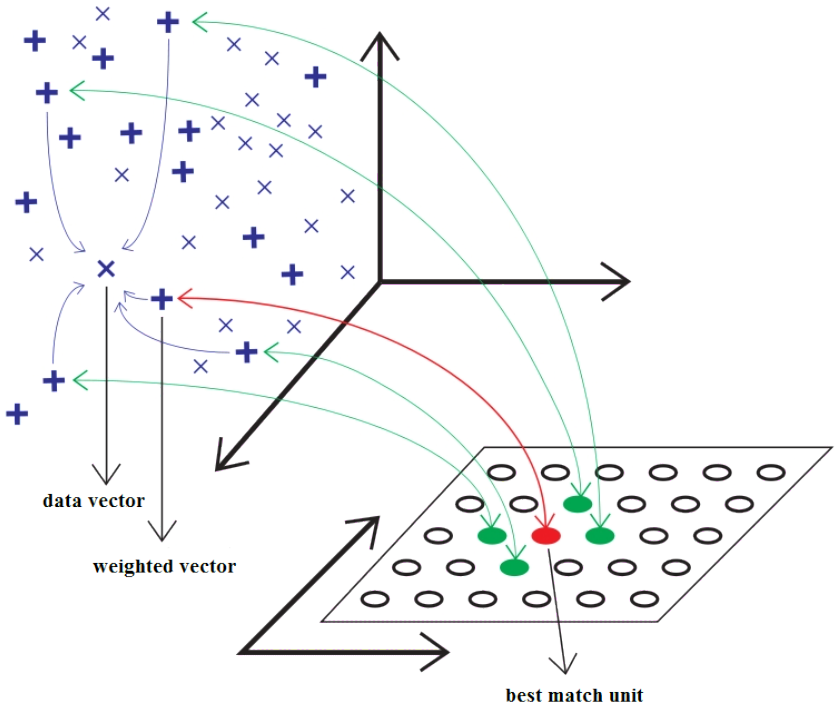
- 1-dimensional, 2-dimensional or 3-dimensional map,
- size of the map (number of neurons),
- map topology or shape: rectangle or hexa,
- a map node (neuron) associated with a weight vector (prototype),
- neighbor nodes are linked by topological links.



# Clustering algorithms: Self-organizing Map

Assignment: find the nearest centroid (bmu)

$$k = \arg \min_{k=1,..,K} \|\mathbf{x}_i - \mathbf{w}_k\|^2$$



# Clustering algorithms: Self-organizing Map

Update: "Winner-take-most" rule for adjusting the prototypes

- $\mathbf{w}_0$ is the best match unit of the current $\mathbf{x}_i$ assigned to $\mathbf{w}_0$
- $\mathbf{w}_0 = \mathbf{w}_0 + \alpha(\mathbf{x}_i - \mathbf{w}_0)$
- for all nodes $\mathbf{w}_r$, the neighbors of $\mathbf{w}_0$, $\mathbf{w}_r = \mathbf{w}_r + \beta(\mathbf{x}_i - \mathbf{w}_0)$ where $\alpha, \beta$ are converging constants ($\alpha > \beta$) or neighborhood functions (kernel functions)

# Clustering algorithms: Self-organizing Map

Advantages:

- Reduction of dimensionality to visualize data in a low dimensional space.
- Capable of clustering large, complex data sets.
- Neighborhood functions help the algorithm convergence.

# Clustering algorithms: Self-organizing Map

Advantages:

- Reduction of dimensionality to visualize data in a low dimensional space.
- Capable of clustering large, complex data sets.
- Neighborhood functions help the algorithm convergence.

Variants:

- Neural Gas
- Growing Neural Gas
- etc...

# Clustering evaluation

Each cluster $C_k$ is associated with:

- a centroid (prototype): $\mathbf{w}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \mathbf{x}_i$
- intra-class variance (quantization error):
  $error = \sum_{i=1}^{n_k} d(\mathbf{x}_i, \mathbf{w}_k)$
- inter-class variance: $error = \sum_{k=1}^{K} \sum_{l \neq k; l=1}^{K} \sum_{j=1}^{n_l} d(\mathbf{x}_i, \mathbf{w}_k)$

# Clustering evaluation

## Internal validation:

- Davies-Bouldin index measures the correlation between two clusters:

$$DB = \frac{1}{N}\sum_{i=1}^{K} \max_{j,i\neq j} \frac{\mu_i + \mu_j}{d(\mathbf{w}_i, \mathbf{w}_j)}$$

where $\mu_i = \frac{1}{n_i}\sum_{j=1}^{n_i} d(\mathbf{w}_i, \mathbf{x}_j)$

- Dunn index measures dense and well-separated clusters:

$$Dunn = \min_{i=1,..,K}\Big(\min_{j=1,..,K;i\neq j}\Big(\frac{d(\mathbf{w}_i, \mathbf{w}_j)}{\max_{l=1,..,K}\Delta(l)}\Big)\Big)$$

where $\Delta(l)$ measures the intra-cluster distance of cluster l

# Clustering evaluation

Clustering methods group these objects into $K$ clusters, thus two partitions to compare are defined: $\mathcal{C} = \{C_1, .., C_K\}$ is a random variable for data cluster assignments $\mathcal{X} = \{\mathbf{x}_1, .., \mathbf{x}_N\}$, and $Y = \{y_1, .., y_N\}$, where $y_l \in \mathcal{B} = \{B_1, .., B_L\}$ is a variable for the original labels.

| $\mathcal{B}\backslash\mathcal{C}$ | $C_1$ | $C_2$ | $\cdots$ | $C_k$ | $\cdots$ | $C_K$ | Sum |
|---|---|---|---|---|---|---|---|
| $B_1$ | $n_{11}$ | $n_{12}$ | $\cdots$ | $n_{1k}$ | $\cdots$ | $n_{1K}$ | $n_{B_1}$ |
| $B_2$ | $n_{21}$ | $n_{22}$ | $\cdots$ | $n_{2k}$ | $\cdots$ | $n_{2K}$ | $n_{B_2}$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\ddots$ | $\vdots$ | $\ddots$ | $\vdots$ | $\vdots$ |
| $B_l$ | $n_{l1}$ | $n_{l2}$ | $\cdots$ | $n_{lk}$ | $\cdots$ | $n_{lK}$ | $n_{B_l}$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\ddots$ | $\vdots$ | $\ddots$ | $\vdots$ | $\vdots$ |
| $B_L$ | $n_{L1}$ | $n_{L2}$ | $\cdots$ | $n_{Lk}$ | $\cdots$ | $n_{LK}$ | $n_{B_L}$ |
| Sum | $n_{C_1}$ | $n_{C_2}$ | | $n_{C_k}$ | | $n_{C_K}$ | $N$ |

# Clustering evaluation

## External validation:

- Accuracy reflects the proportion of objects that were correctly assigned:

$$Acc = \frac{1}{N}\sum_{k=1}^{K}\max_{l=1,..,L}(n_{lk})$$

- Mutual Information measures how much information is shared between a clustering and a ground-truth classification:

$$MI = \sum_{l=1}^{L}\sum_{k=1}^{K} n_{lk} log_2(\frac{Nn_{lk}}{n_{B_l} n_{C_k}})$$

- Normalized Mutual Information:

$$NMI = \frac{MI}{\sqrt{(\sum_{l=1}^{L} n_{B_l} log_2(\frac{n_{B_l}}{N}))(\sum_{k=1}^{K} n_{C_k} log_2(\frac{n_{C_k}}{N}))}}$$

# Clustering evaluation

External validation:

- Rand index computes how similar the obtained clusters are to the benchmark classifications: $Rand = \frac{N_{00}+N_{11}}{N_{00}+N_{01}+N_{10}+N_{11}}$
- Jaccard index is used to quantify the similarity between two partitions: $Jaccard = \frac{N_{11}}{N_{00}+N_{10}+N_{01}}$

where
$N_{11}$ is the number of data pairs in the same cluster in both $\mathcal{B}$ and $\mathcal{C}$;
$N_{10}$ is the number of data pairs in the same cluster in $\mathcal{B}$ but not $\mathcal{C}$;
$N_{01}$ is the number of data pairs in the same cluster in $\mathcal{C}$ but not $\mathcal{B}$;
$N_{00}$ is the number of data pairs in different clusters in both $\mathcal{B}$ and $\mathcal{C}$.

# Clustering evaluation

Example: 10 objects $\in \mathcal{X}$ having the original labels $Y = \{1, 1, 1, 1, 2, 2, 2, 2, 2, 2\}$ are clustered into 3 clusters, $Y_{new} = \{1, 1, 1, 2, 2, 2, 3, 3, 3, 3\}$. We have:

- $B_1 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$ and $B_2 = \{\mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_7, \mathbf{x}_8, \mathbf{x}_9, \mathbf{x}_{10}\}$
- $C_1 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$, $C_2 = \{\mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$ and $C_3 = \{\mathbf{x}_7, \mathbf{x}_8, \mathbf{x}_9, \mathbf{x}_{10}\}$

| $\mathcal{B}\backslash\mathcal{C}$ | $C_1$ | $C_2$ | $C_3$ | Sum |
|---|---|---|---|---|
| $B_1$ | 3 | 1 | 0 | 4 |
| $B_2$ | 0 | 2 | 4 | 6 |
| Sum | 3 | 3 | 4 | 10 |

# Clustering evaluation

| $\mathcal{B}\backslash\mathcal{C}$ | $C_1$ | $C_2$ | $C_3$ | Sum |
|---|---|---|---|---|
| $B_1$ | 3 | 1 | 0 | 4 |
| $B_2$ | 0 | 2 | 4 | 6 |
| Sum | 3 | 3 | 4 | 10 |

- Accuracy: $Acc = \frac{1}{N}\sum_{k=1}^{K} \max_{l=1,..,L}(n_{lk}) = \frac{1}{10}\sum_{k=1}^{3} \max((3,0),(1,2),(0,4)) = \frac{3+2+4}{10} = 0.9$
- Mutual Information: $MI = \sum_{l=1}^{L}\sum_{k=1}^{K} n_{lk} log_2(\frac{Nn_{lk}}{n_{B_l} n_{C_k}}) = 3log_2(\frac{10*3}{3*4}) + 1log_2(\frac{10*1}{3*4}) + 2log_2(\frac{10*2}{3*6}) + 4log_2(\frac{10*4}{4*6}) =$

# Clustering evaluation

| $\mathcal{B}\backslash\mathcal{C}$ | $C_1$ | $C_2$ | $C_3$ | Sum |
|---|---|---|---|---|
| $B_1$ | 3 | 1 | 0 | 4 |
| $B_2$ | 0 | 2 | 4 | 6 |
| Sum | 3 | 3 | 4 | 10 |

- Accuracy: $Acc = \frac{1}{N}\sum_{k=1}^{K} \max_{l=1,..,L}(n_{lk}) = \frac{1}{10}\sum_{k=1}^{3} \max((3,0),(1,2),(0,4)) = \frac{3+2+4}{10} = 0.9$
- Mutual Information: $MI = \sum_{l=1}^{L}\sum_{k=1}^{K} n_{lk} log_2(\frac{Nn_{lk}}{n_{B_l} n_{C_k}}) = 3log_2(\frac{10*3}{3*4}) + 1log_2(\frac{10*1}{3*4}) + 2log_2(\frac{10*2}{3*6}) + 4log_2(\frac{10*4}{4*6}) = 6.954$
- Normalized Mutual Information: $NMI = \frac{MI}{\sqrt{(\sum_{l=1}^{L} n_{B_l} log_2(\frac{n_{B_l}}{N}))(\sum_{k=1}^{K} n_{C_k} log_2(\frac{n_{C_k}}{N}))}} = \frac{MI}{\sqrt{(3log_2(\frac{3}{10})+3log_2(\frac{3}{10})+4log_2(\frac{4}{10}))(4log_2(\frac{4}{10})+6log_2(\frac{6}{10}))}} =$

# Clustering evaluation

| $\mathcal{B}\backslash\mathcal{C}$ | $C_1$ | $C_2$ | $C_3$ | Sum |
|---|---|---|---|---|
| $B_1$ | 3 | 1 | 0 | 4 |
| $B_2$ | 0 | 2 | 4 | 6 |
| Sum | 3 | 3 | 4 | 10 |

- Accuracy: $Acc = \frac{1}{N}\sum_{k=1}^{K} \max_{l=1,..,L}(n_{lk}) = \frac{1}{10}\sum_{k=1}^{3} \max((3,0),(1,2),(0,4)) = \frac{3+2+4}{10} = 0.9$
- Mutual Information: $MI = \sum_{l=1}^{L}\sum_{k=1}^{K} n_{lk} log_2(\frac{Nn_{lk}}{n_{B_l} n_{C_k}}) = 3log_2(\frac{10*3}{3*4}) + 1log_2(\frac{10*1}{3*4}) + 2log_2(\frac{10*2}{3*6}) + 4log_2(\frac{10*4}{4*6}) = 6.954$
- Normalized Mutual Information: $NMI = \frac{MI}{\sqrt{(\sum_{l=1}^{L} n_{B_l} log_2(\frac{n_{B_l}}{N}))(\sum_{k=1}^{K} n_{C_k} log_2(\frac{n_{C_k}}{N}))}} = \frac{MI}{\sqrt{(3log_2(\frac{3}{10})+3log_2(\frac{3}{10})+4log_2(\frac{4}{10}))(4log_2(\frac{4}{10})+6log_2(\frac{6}{10}))}} = \frac{6.954}{\sqrt{152.531}} = 0.563$

# Clustering evaluation

- $B_1 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$ and $B_2 = \{\mathbf{x}_5, \mathbf{x}_6, \mathbf{x}_7, \mathbf{x}_8, \mathbf{x}_9, \mathbf{x}_{10}\}$
- $C_1 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$, $C_2 = \{\mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$ and $C_3 = \{\mathbf{x}_7, \mathbf{x}_8, \mathbf{x}_9, \mathbf{x}_{10}\}$

$\rightarrow N_{00} = 22$, $N_{11} = 10$, $N_{10} = 11$ and $N_{01} = 2$

- Rand index: $Rand = \frac{N_{00}+N_{11}}{\frac{N}{2}} = \frac{22+10}{22+10+11+2} = \frac{32}{45} = 0.711$
- Jaccard index:
  $Jaccard = \frac{N_{11}}{N_{11}+N_{10}+N_{01}} = \frac{10}{10+11+2} = \frac{10}{23} = 0.434$

# Conclusion

- Clustering: unsupervised learning.
- Grouping the homogeneous objects into clusters (low intra-class variance and high inter-class variance).
- It exists many criteria to evaluate a clustering (internal and external validation).
- Algorithms are rich: AHC, $K$-means, EM et many mores.