Artificial Intelligence for Medicine II

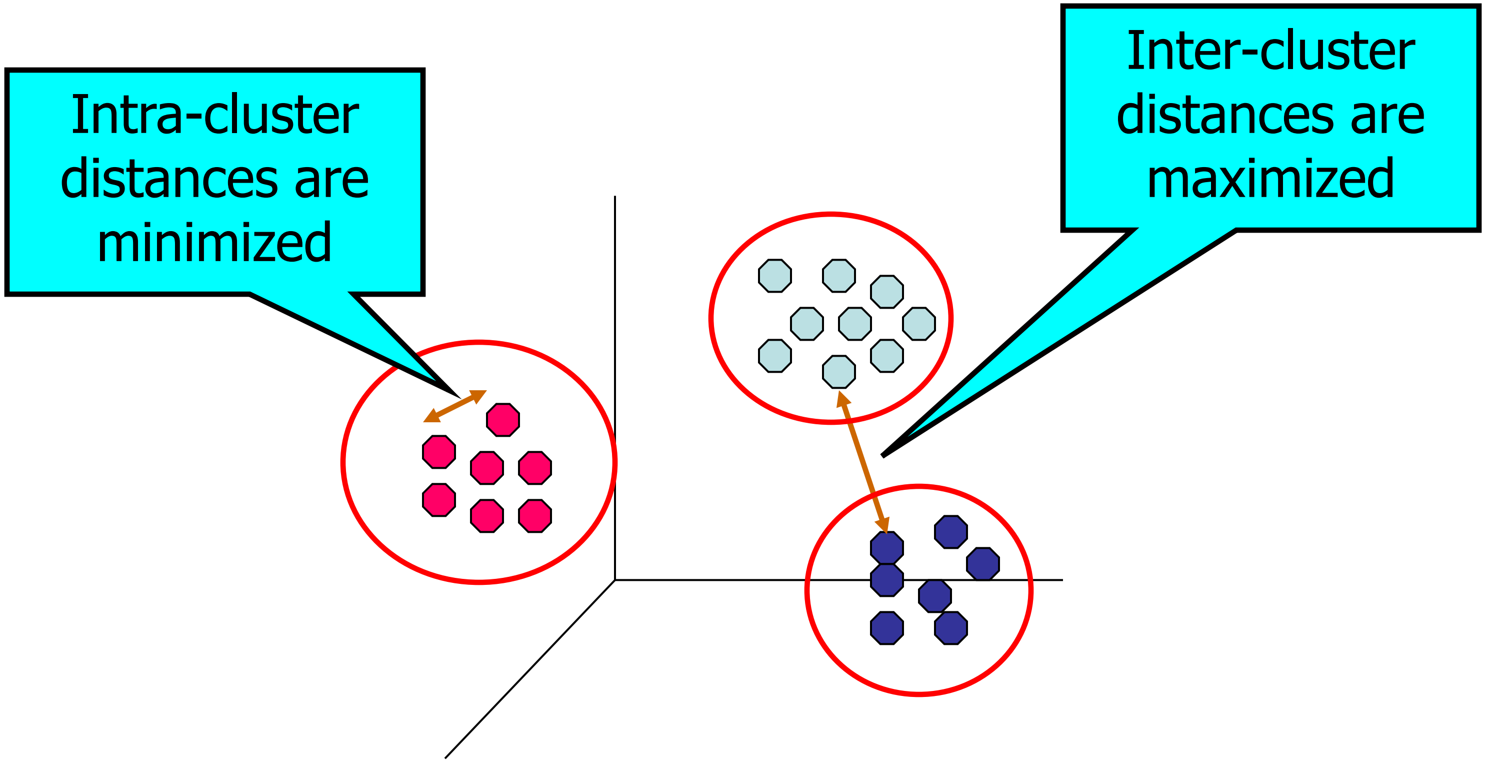
Spring 2025

Lecture 81: Unsupervised Learning Clustering

(Many slides adapted from Bing Liu, Han, Kamber & Pei; Tan, Steinbach, Kumar   and the web)

What is Cluster Analysis?

- Given a set of objects, place them in groups such that the objects in a group are similar (or related) to one another and different from (or unrelated to) the objects in other groups



Intra-cluster distances are minimized

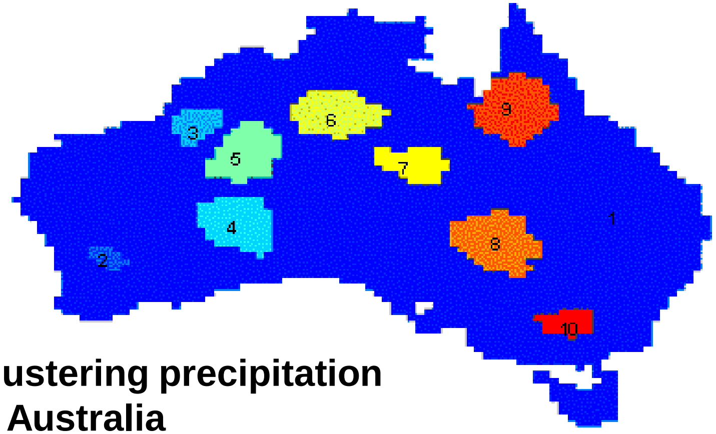
Inter-cluster distances are maximized

Applications of Cluster Analysis

- **Understanding**
  - Group related documents for browsing, group genes and proteins that have similar functionality, or group stocks with similar price fluctuations

	Discovered Clusters	*Industry Group*
1	Applied-Matl-DOWN,Bay-Network-Down,3-COM-DOWN, Cabletron-Sys-DOWN,CISCO-DOWN,HP-DOWN, DSC-Comm-DOWN,INTEL-DOWN,LSI-Logic-DOWN, Micron-Tech-DOWN,Texas-Inst-Down,Tellabs-Inc-Down, Natl-Semiconduct-DOWN,Oracl-DOWN,SGI-DOWN, Sun-DOWN	Technology1-DOWN
2	Apple-Comp-DOWN,Autodesk-DOWN,DEC-DOWN, ADV-Micro-Device-DOWN,Andrew-Corp-DOWN, Computer-Assoc-DOWN,Circuit-City-DOWN, Compaq-DOWN, EMC-Corp-DOWN, Gen-Inst-DOWN, Motorola-DOWN,Microsoft-DOWN,Scientific-Atl-DOWN	Technology2-DOWN
3	Fannie-Mae-DOWN,Fed-Home-Loan-DOWN, MBNA-Corp-DOWN,Morgan-Stanley-DOWN	Financial-DOWN
4	Baker-Hughes-UP,Dresser-Inds-UP,Halliburton-HLD-UP, Louisiana-Land-UP,Phillips-Petro-UP,Unocal-UP, Schlumberger-UP	Oil-UP

- **Summarization**
  - Reduce the size of large data sets



Clustering precipitation in Australia

Clustering Problem Formally

- Given
  - a database $D=\{t_1,t_2,\ldots,t_n\}$ of tuples and
  - an integer value k,
- the *Clustering Problem* is to define
  - a mapping $f:D \rightarrow \{1,..,k\}$
    where each t_i is assigned to one cluster K_j, $1<=j<=k$.
- A *cluster*, K_j, contains precisely those tuples mapped to it.
- Unlike classification problem, clusters are not known *a priori*.

General Applications of Clustering

- Pattern Recognition
- Spatial Data Analysis
  - create thematic maps in GIS by clustering feature spaces
  - detect spatial clusters and explain them in spatial data mining
- Image Processing
- Economic Science (especially market research)
- WWW
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns
- Medicine
  - Patient Stratification (Group patients with similar chatracteristics)
  - Heathcare data analysis
  - Medical Imaging (segmentation and detecting anomalies)

Examples of Clustering Applications

- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- Land use: Identification of areas of similar land use in an earth observation database
- Insurance: Identifying groups of motor insurance policy holders with a high average claim cost
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Earth-quake studies: Observed earth quake epicenters should be clustered along continent faults

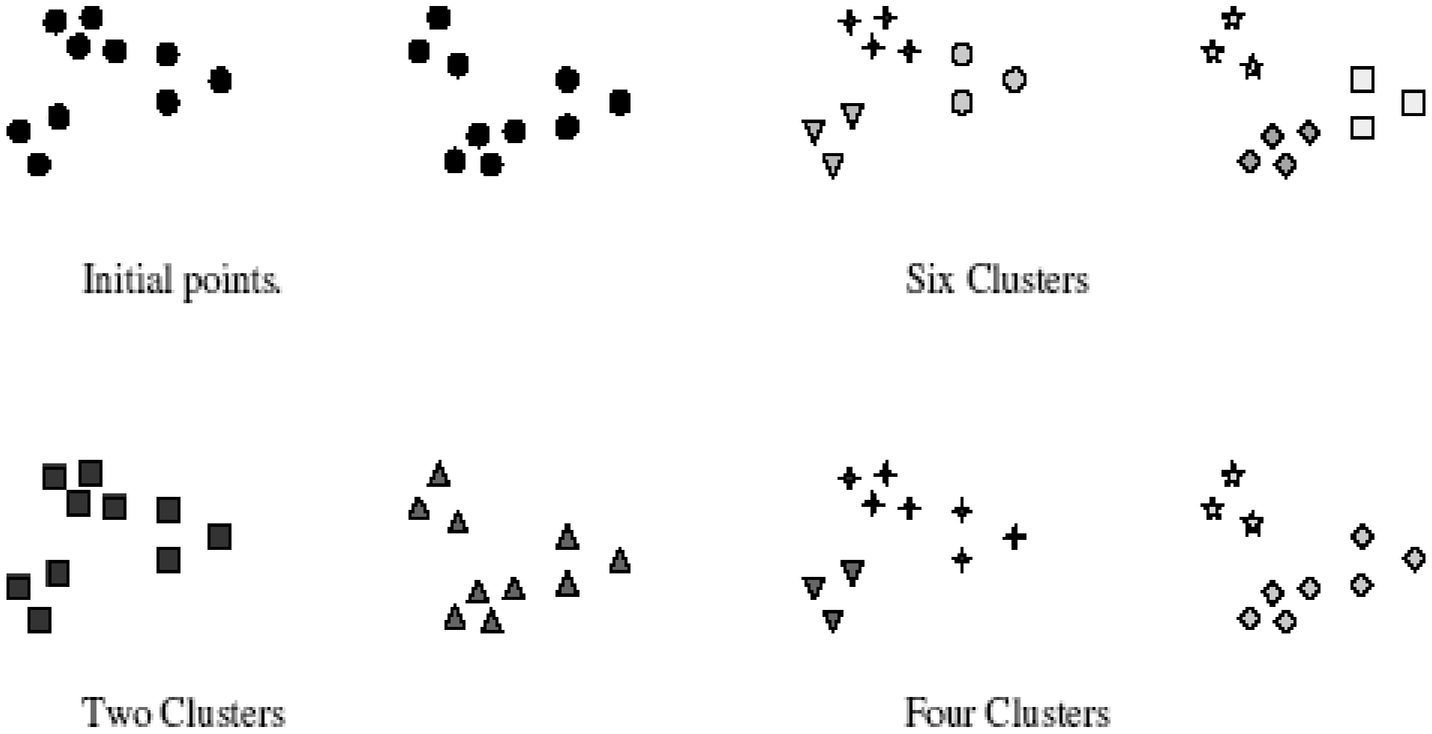
Clustering as a Preprocessing Tool (Utility)

- Summarization:
  - Preprocessing for regression, PCA, classification, and association analysis
- Compression:
  - Image processing: vector quantization
- Finding K-nearest Neighbors
  - Localizing search to one or a small number of clusters
- Outlier detection
  - Outliers are often viewed as those “far away” from any cluster

Clustering Issues

- The appropriate number of clusters for each data set.
- How to define similarity or the criterion used to group data together.
- Outlier handling is difficult. Should they be a part of an existing cluster, or another cluster?
- Dynamic database, how to update the clusters when there are changes in data.
- The semantic meaning of each cluster. (Contrast with classes in classification process, each has a definitive meaning.)
- Type of attributes that the clustering algorithm can handle.
- Scalability to large datasets.

Notion of a cluster is ambigious



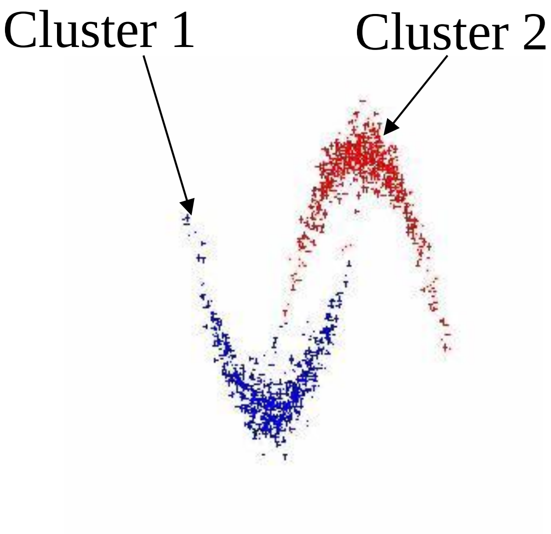
Initial points.

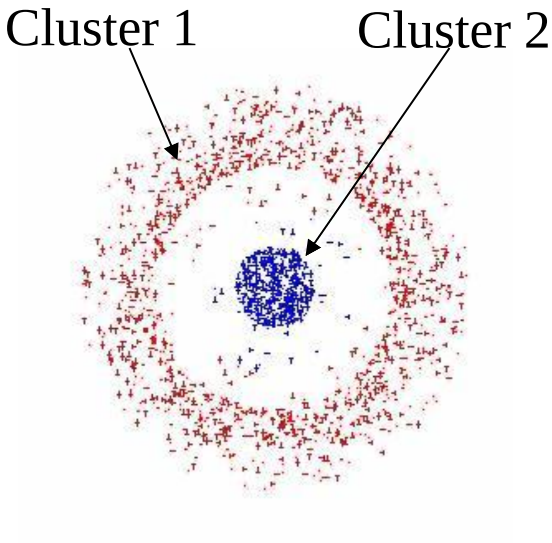
Six Clusters

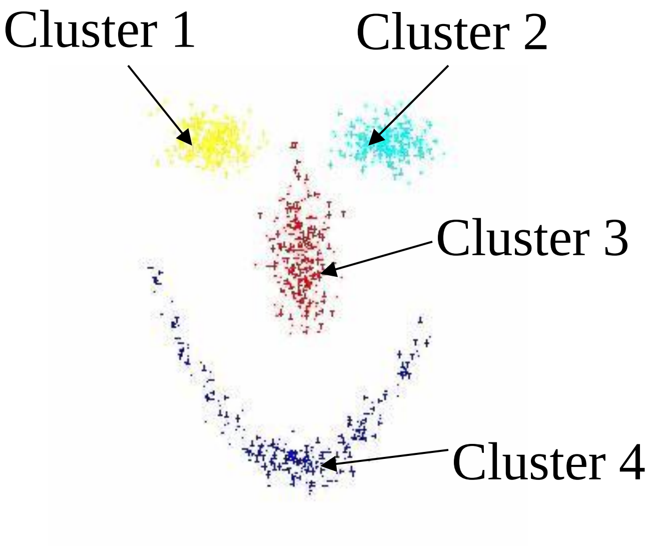
Two Clusters

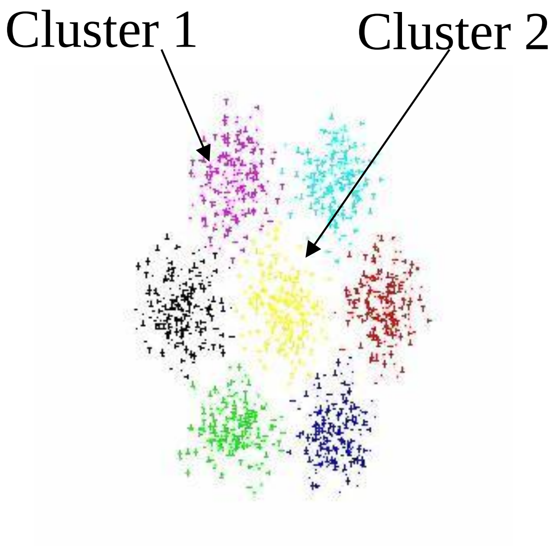
Four Clusters

Different types of clusters









Quality: What Is Good Clustering?

- A good clustering method will produce high quality clusters
  - high intra-class similarity: cohesive within clusters
  - low inter-class similarity: distinctive between clusters
- The quality of a clustering method depends on
  - the similarity measure used by the method
  - its implementation, and
  - Its ability to discover some or all of the hidden patterns

Requirements and Challenges

- Scalability
  - Clustering all the data instead of only on samples
- Ability to deal with different types of attributes
  - Numerical, binary, categorical, ordinal, linked, and mixture of these
- Constraint-based clustering
  - User may give inputs on constraints
  - Use domain knowledge to determine input parameters
- Interpretability and usability
- Others
  - Discovery of clusters with arbitrary shape
  - Ability to deal with noisy data
  - Incremental clustering and insensitivity to input order
  - High dimensionality

Measure the Quality of Clustering

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

Similarity and Dissimilarity Metric

- Similarity
  - Numerical measure of how alike two data objects are.
  - Is higher when objects are more alike.
  - Often falls in the range [0,1]
- Dissimilarity
  - Numerical measure of how different two data objects are.
  - Is lower when objects are more alike.
  - Minimum dissimilarity is often 0.
  - Upper limit varies
- Proximity refers to a similarity or dissimilarity

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*:

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

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

- If *q* = *1*, *d* is Manhattan distance

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + ... + |x_{ip} - x_{jp}|$$

Similarity and Dissimilarity Between Objects (Cont.)

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

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

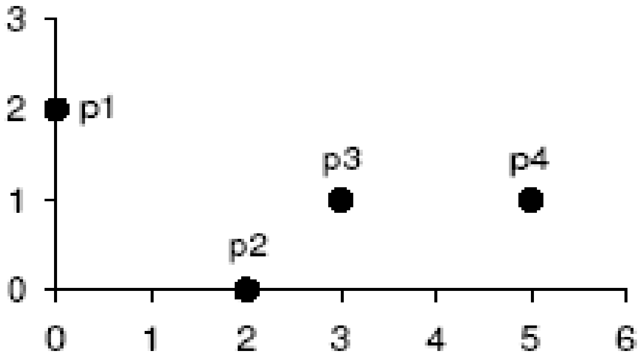
  – Properties

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

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

Euclidean Distance

Euclidean Distance



point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix

Source : S. Ranka

Other Similarity and Dissimilarity Metrics Between Objects

$$\text{Dice}: sim(t_i, t_j) = \frac{2\sum_{h=1}^{k} t_{ih} t_{jh}}{\sum_{h=1}^{k} t_{ih}^2 + \sum_{h=1}^{k} t_{jh}^2}$$

$$\text{Jaccard}: sim(t_i, t_j) = \frac{\sum_{h=1}^{k} t_{ih} t_{jh}}{\sum_{h=1}^{k} t_{ih}^2 + \sum_{h=1}^{k} t_{jh}^2 - \sum_{h=1}^{k} t_{ih} t_{jh}}$$

$$\text{Cosine}: sim(t_i, t_j) = \frac{\sum_{h=1}^{k} t_{ih} t_{jh}}{\sqrt{\sum_{h=1}^{k} t_{ih}^2 \sum_{h=1}^{k} t_{jh}^2}}$$

$$\text{Overlap}: sim(t_i, t_j) = \frac{\sum_{h=1}^{k} t_{ih} t_{jh}}{\min\left(\sum_{h=1}^{k} t_{ih}^2, \sum_{h=1}^{k} t_{jh}^2\right)}$$

$$t_i = \langle t_{i1}, \ldots, t_{ik} \rangle, t_j = \langle t_{j1}, \ldots, t_{jk} \rangle$$

Data Structures

- **Data matrix**
  - This represents n objects, such as persons, with p variables (also called measurements or attributes), such as age, height, gender, race, and so on.
  - Called “two modes” : since rows and columns represent different entities

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

- **Proximity (Similarity/Dissimilarity) matrix**
  - Stores a collection of proximities that are available for all pairs of n objects. (n by n matrix)
  - Called “one mode” : since it reprsents the same entity
  - d(i,j) is the measured difference or dissimilarity between objects i and j.

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

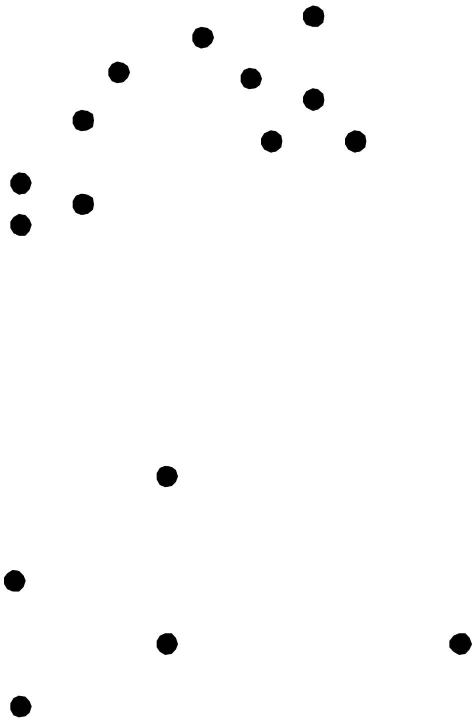
Considerations for Cluster Analysis

- Partitioning criteria
  - Single level vs. hierarchical partitioning (often, multi-level hierarchical partitioning is desirable)
- Separation of clusters
  - Exclusive (e.g., one customer belongs to only one region) vs. non-exclusive (e.g., one document may belong to more than one class)
- Similarity measure
  - Distance-based (e.g., Euclidian, road network, vector) vs. connectivity-based (e.g., density or contiguity)
- Clustering space
  - Full space (often when low dimensional) vs. subspaces (often in high-dimensional clustering)

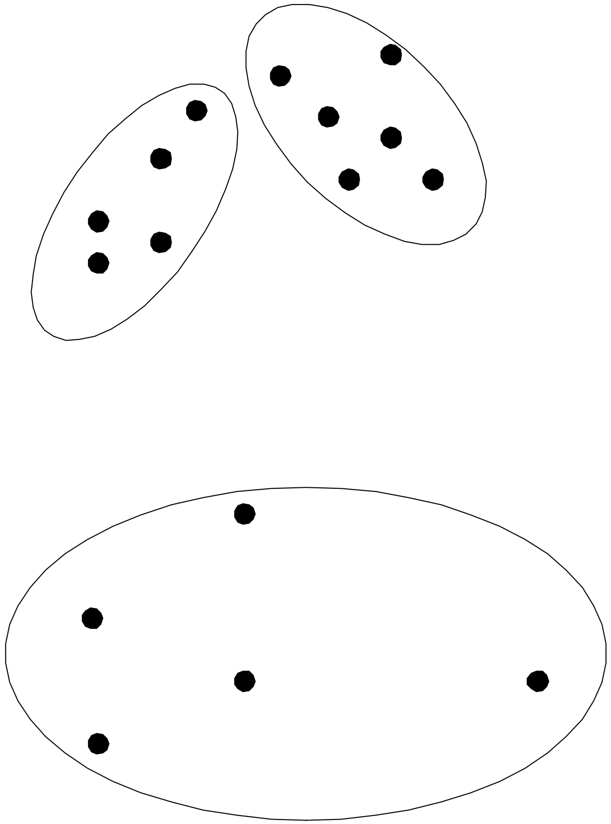
Types of Clusterings

- A clustering is a set of clusters
- Important distinction between hierarchical and partitional sets of clusters
  - Partitional Clustering
    - A division of data objects into non-overlapping subsets (clusters)
  - Hierarchical clustering
    - A set of nested clusters organized as a hierarchical tree

Partitional Clustering

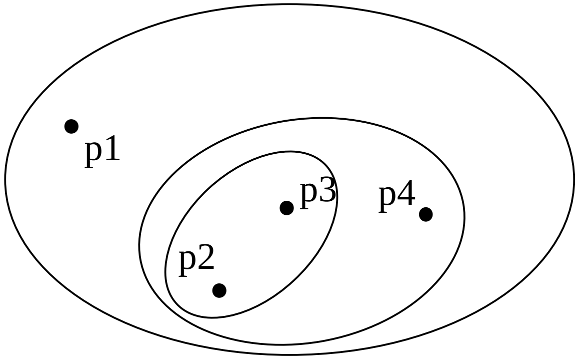


Original Points

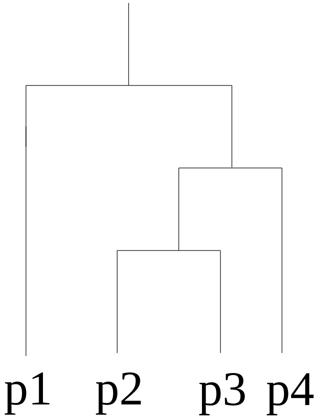


A Partitional Clustering

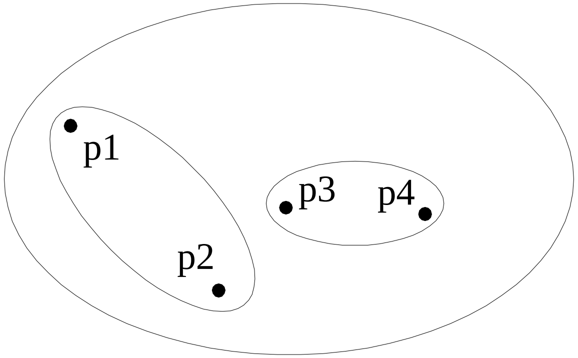
Hierarchical Clustering



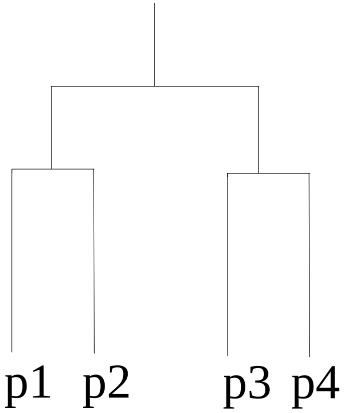
Traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Hierarchical Clustering



Non-traditional Dendrogram

Other Distinctions Between Sets of Clusters

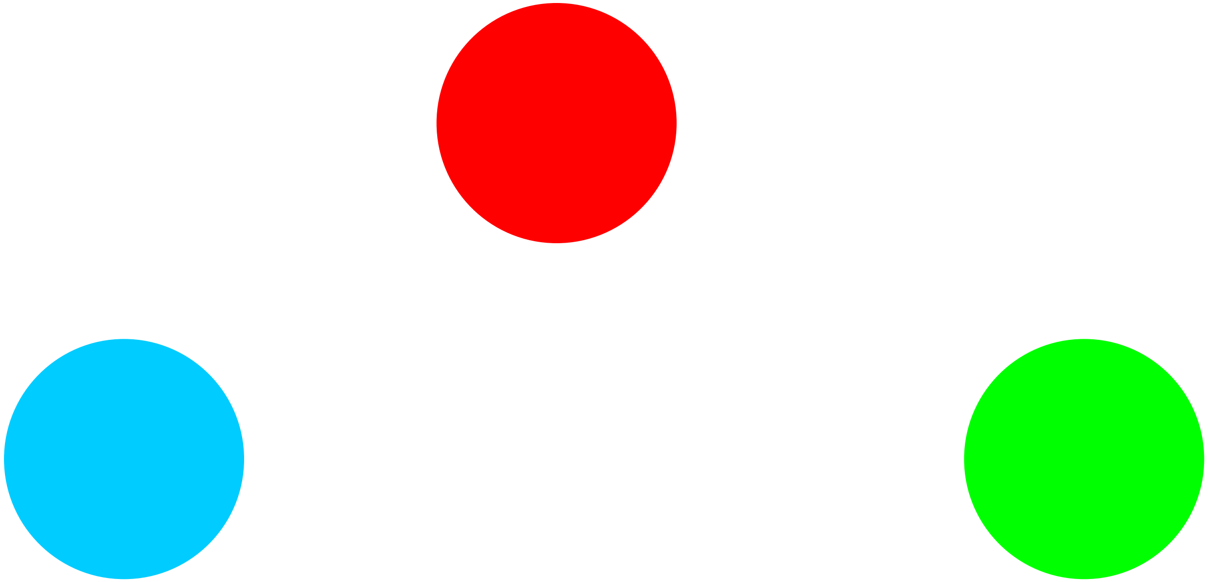
- Exclusive versus non-exclusive
  - In non-exclusive clusterings, points may belong to multiple clusters.
    - Can belong to multiple classes or could be ‘border’ points
  - Fuzzy clustering (one type of non-exclusive)
    - In fuzzy clustering, a point belongs to every cluster with some weight between 0 and 1
    - Weights must sum to 1
    - Probabilistic clustering has similar characteristics
- Partial versus complete
  - In some cases, we only want to cluster some of the data

Types of Clusters

- Well-separated clusters
- Prototype-based clusters
- Contiguity-based clusters
- Density-based clusters
- Described by an Objective Function

Types of Clusters: Well-Separated

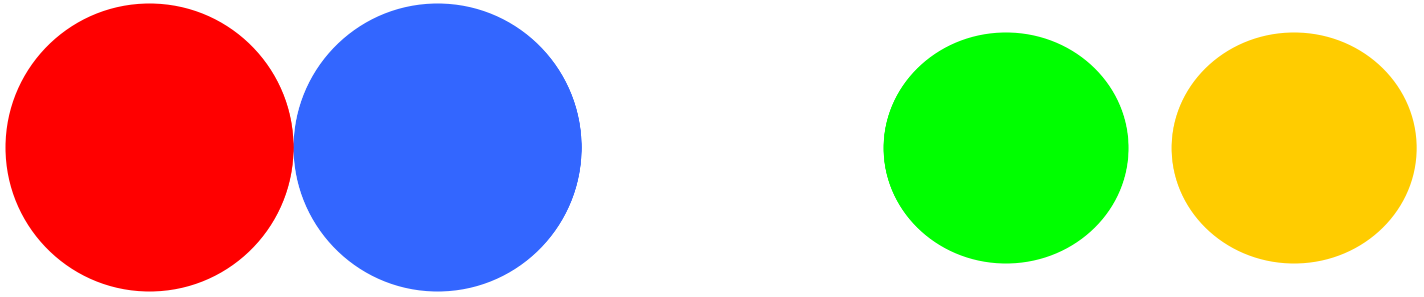
- Well-Separated Clusters:
  - A cluster is a set of points such that any point in a cluster is closer (or more similar) to every other point in the cluster than to any point not in the cluster.



3 well-separated clusters

Types of Clusters: Prototype-Based

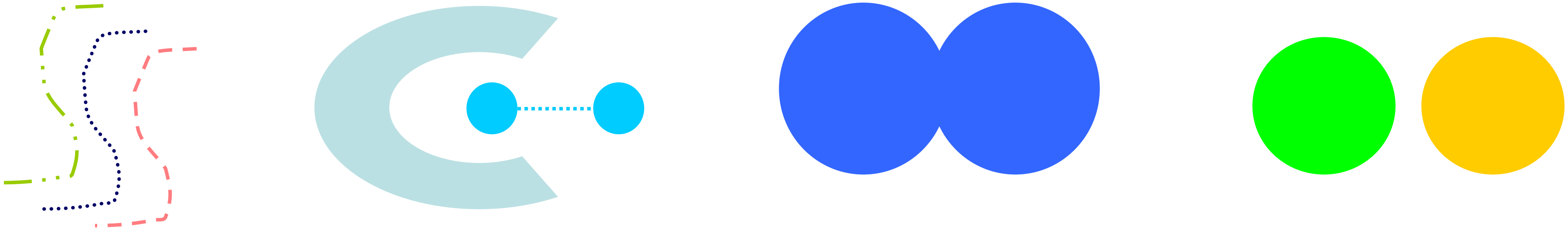
- Prototype-based
  - A cluster is a set of objects such that an object in a cluster is closer (more similar) to the prototype or "center" of a cluster, than to the center of any other cluster
  - The center of a cluster is often a centroid, the average of all the points in the cluster, or a medoid, the most "representative" point of a cluster



4 center-based clusters

Types of Clusters: Contiguity-Based

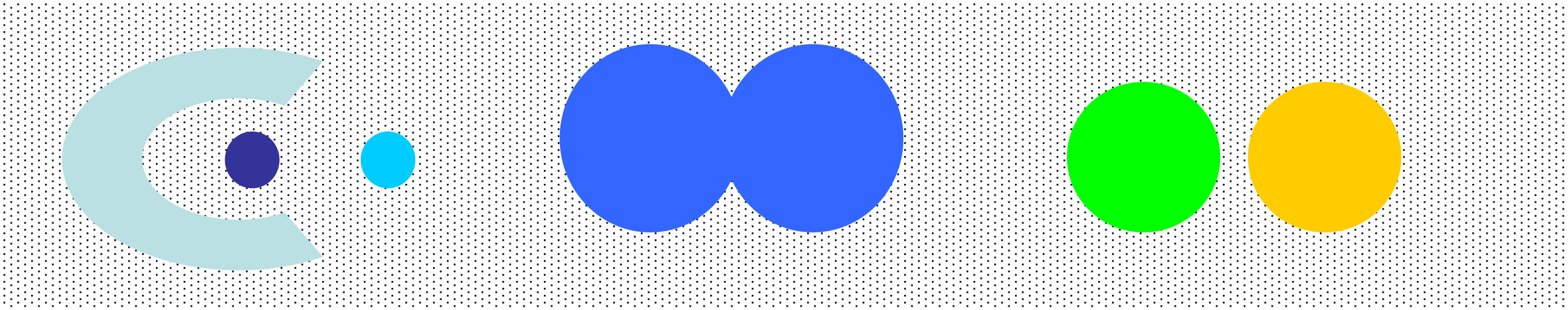
- Contiguous Cluster (Nearest neighbor or Transitive)
  - A cluster is a set of points such that a point in a cluster is closer (or more similar) to one or more other points in the cluster than to any point not in the cluster.



8 contiguous clusters

Types of Clusters: Density-Based

- Density-based
  - A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density.
  - Used when the clusters are irregular or intertwined, and when noise and outliers are present.



6 density-based clusters

Types of Clusters: Objective Function

- Clusters Defined by an Objective Function
  - Finds clusters that minimize or maximize an objective function.
  - Enumerate all possible ways of dividing the points into clusters and evaluate the `goodness' of each potential set of clusters by using the given objective function. (NP Hard)
  - Can have global or local objectives.
    - Hierarchical clustering algorithms typically have local objectives
    - Partitional algorithms typically have global objectives
  - A variation of the global objective function approach is to fit the data to a parameterized model.
    - Parameters for the model are determined from the data.
    - Mixture models assume that the data is a ‘mixture' of a number of statistical distributions.

Characteristics of the Input Data Are Important

- Type of proximity or density measure
  - Central to clustering
  - Depends on data and application

- Data characteristics that affect proximity and/or density are
  - Dimensionality
    - Sparseness
  - Attribute type
  - Special relationships in the data
    - For example, autocorrelation
  - Distribution of the data

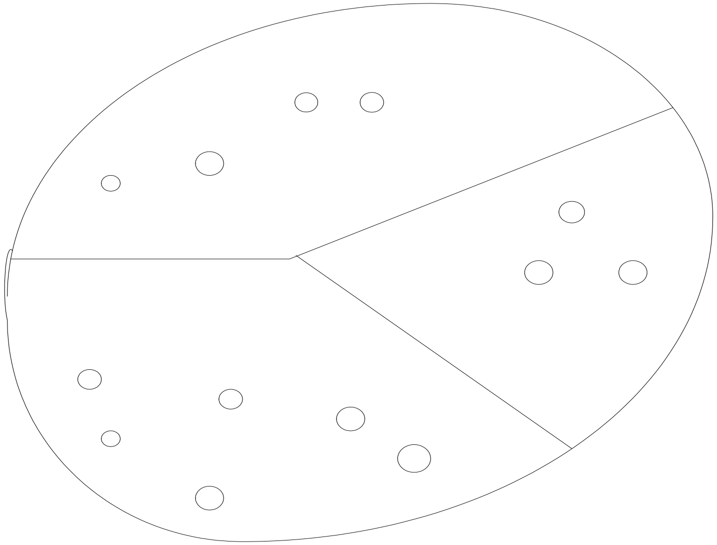
- Noise and Outliers
  - Often interfere with the operation of the clustering algorithm
- Clusters of differing sizes, densities, and shapes

Clustering Algorithms

- K-means and its variants (Partitional algorithms)
- Hierarchical clustering
- Density-based clustering

Partitioning Methods (or algorithms)

- A partitioning method constructs k clusters. It classifies the data into k groups, which together satisfy the requirements of a partition.Each group must contain at least one object.
- Each object must belong to exactly one group.
- k <= n where k is the number of clusters with n objects.



Partitional algorithms

- Partitional algorithms generate a single partition (or clustering) of the data.
- Hierarchical clusterings are widely used in biological, social science where the representations in taxonomy style are preferred.
- Partitional clusterings are frequently used in engineering, they offer more compact representation.
- Partitional algorithms divide data into k clusters. The value of k may or may not be specified.
- Criteria, such as squared error, are used to evaluate the goodness of each proposed clustering.
- Partitioning method: Partitioning a database ***D*** of ***n*** objects into a set of ***k*** clusters, such that the sum of squared distances is minimized (where c_i is the centroid or medoid of cluster C_i)

$$\text{Squared error} = \sum_{j=1}^{k} \sum_{t_{K_j i} \in K_j} dist(C_{K_j}, t_{K_j i})^2$$

- The choice of criterion function is problem dependent.

Partitional algorithms (cont.)

- Achieving global optimility in partition-based algortihms require the exhaustive enumaretion of all posibile partitions
- Evaluate every possible partition containing k clusters. Select the best partition according to the criterion function. (NP Hard)
- The number of possible partitions is enormous, even when k, n are relatively small.
- Let $S(n,k)$ denote the number of partitions of n points into k clusters.

$$S(n,k) = \frac{1}{k!}\sum_{i=1}^{k}(-1)^{k-i}\binom{k}{i}(i)^n$$

- E.g. $S(10,4) = 34105$; $S(19,4) = 11259666000$.

- Partition-based algortihms use heuristic methods such K-means, K-medoids, or a variation of them for clustering.

Partitioning Algorithms: Basic Concept

- Partitioning method: Construct a partition of a database ***D*** of ***n*** objects into a set of ***k*** clusters
- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - 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

Cluster Parameters

Some characteristic values of the clusters:

$$\text{Centroid} = C_K = \frac{\sum_{i=1}^{n}(t_{Ki})}{n} \quad \text{Radius} = R_K = \sqrt{\frac{\sum_{i=1}^{n}(t_{Ki} - C_K)^2}{n}}$$

$$\text{Diameter} = D_K = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(t_{Ki} - t_{Kj})^2}{n(n-1)}}$$

- **Centroid** is the middle of the cluster. It is not necessary to be a data point. Let C_K denote centroid of cluster K.
- **Radius**: square root of average distance from any point of the cluster to its centroid.
- **Diameter**: square root of average mean squared distance between all pairs of points in the cluster
- **Medoid** is an actual item that is located at the central of the cluster. Let M_K denote medoid of cluster K.

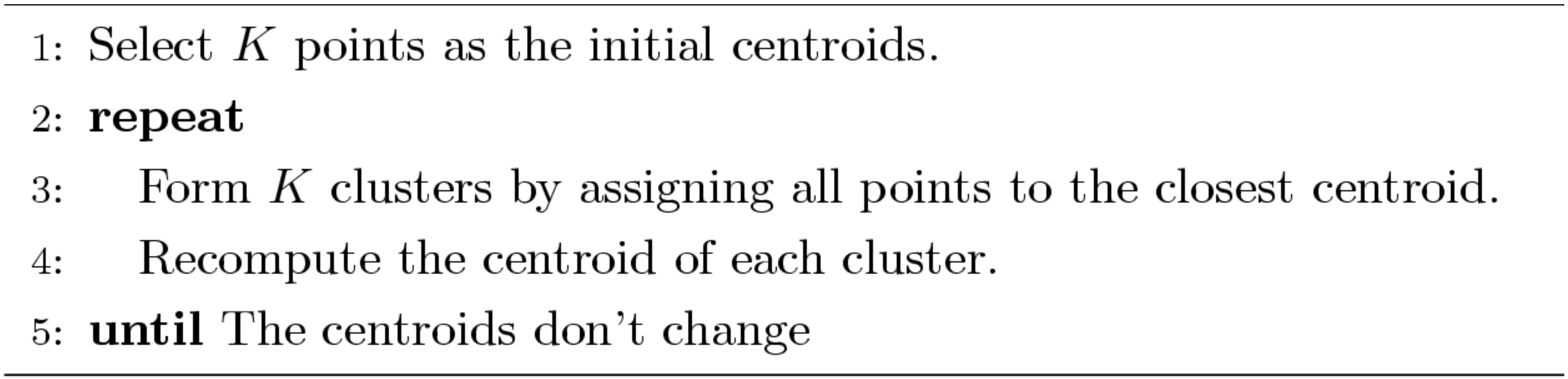
Cluster Parameters

- The squared error for a cluster is the sum of the the squared Euclidean distances between each element in the cluster and the cluster centroid, C_k

$$\boxed{\begin{gathered} se_{K_i} = \sum_{j=1}^{m} \|t_{ij} - C_k\|^2 \\[2em] se_K = \sum_{j=1}^{k} se_{K_j} \end{gathered}}$$

K-means Clustering

- Partitional clustering approach
- Number of clusters, K, must be specified
- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid
- The basic algorithm is very simple

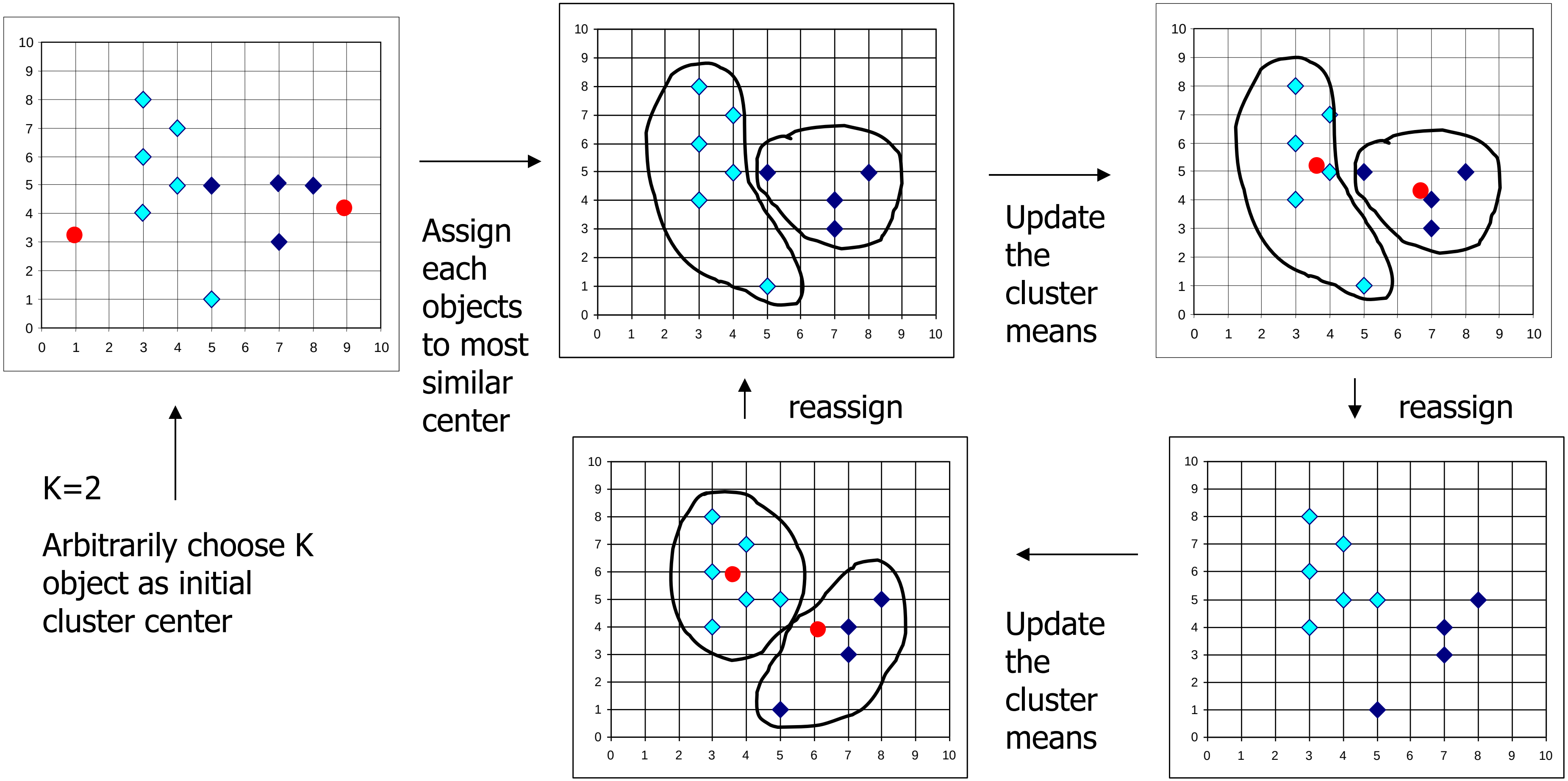


```
Select $K$ points as the initial centroids.
repeat
    Form $K$ clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change
```

The *K-Means* Clustering Method in Detail

- **K-means** clustering: First, pick a numeric attribute to cluster on. Then...
  - **Step 1**: Start with *K* random cluster centers.
  - **Step 2**: Assign each new example to the cluster with the closest center (i.e. closest mean). After all examples have been assigned, calculate the new center (i.e. mean) for each cluster.
  - **Step 3**: Repeat Step 2 until there is convergence. The algorithm may stop when:
    - only a small number of examples are assigned to different clusters, or
    - a fixed number of iterations have been executed,
    - The centroids don’t change or
    - the difference between successive squared errors is below a certain threshold (so called ‘minimum-squared-error clustering)’. The ‘squared error’ is the Euclidian distance between each element in the cluster and the cluster center.

The *K-Means* Clustering Method



K=2

Arbitrarily choose K object as initial cluster center

Assign each objects to most similar center

Update the cluster means

reassign

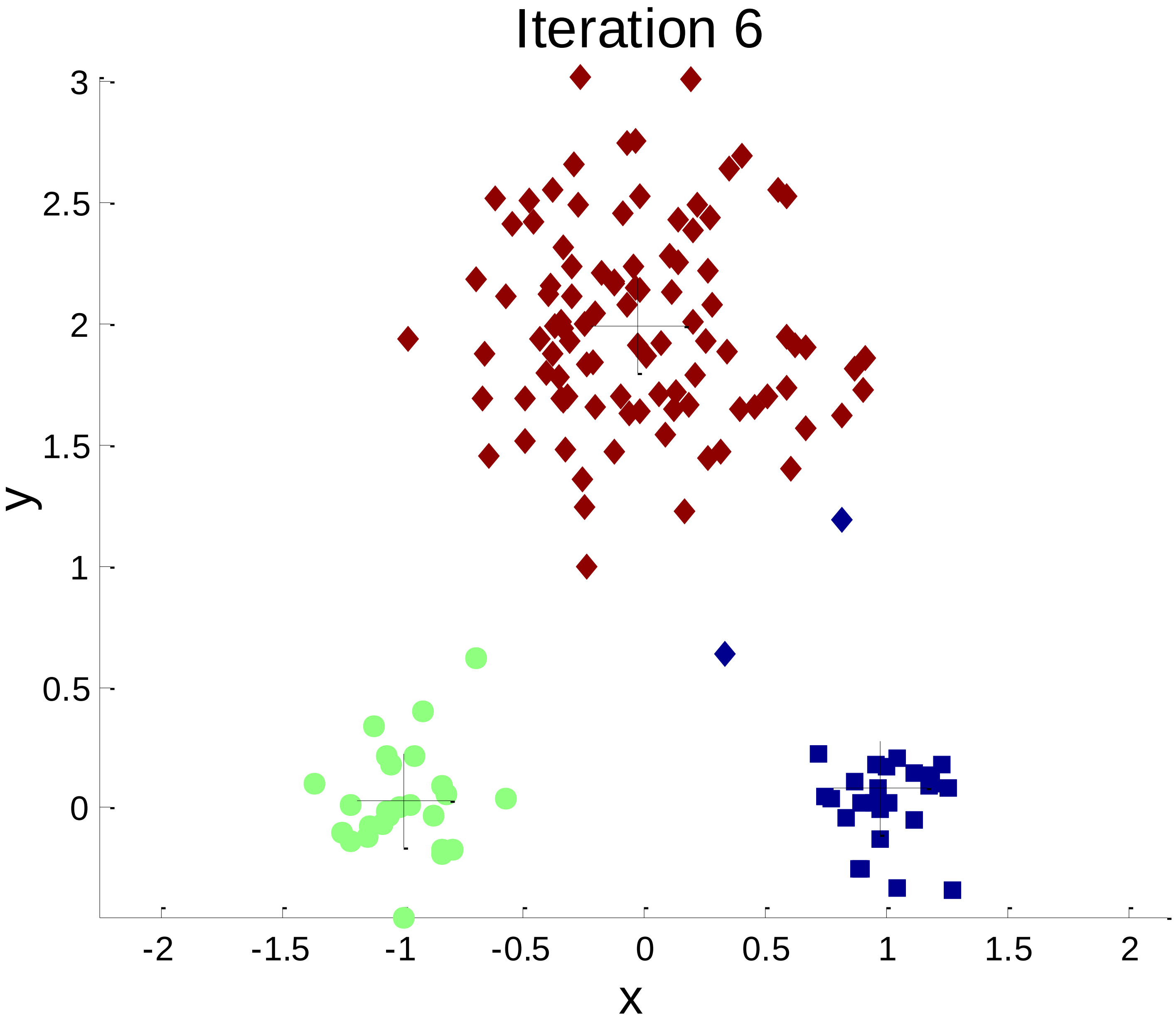
Update the cluster means

reassign

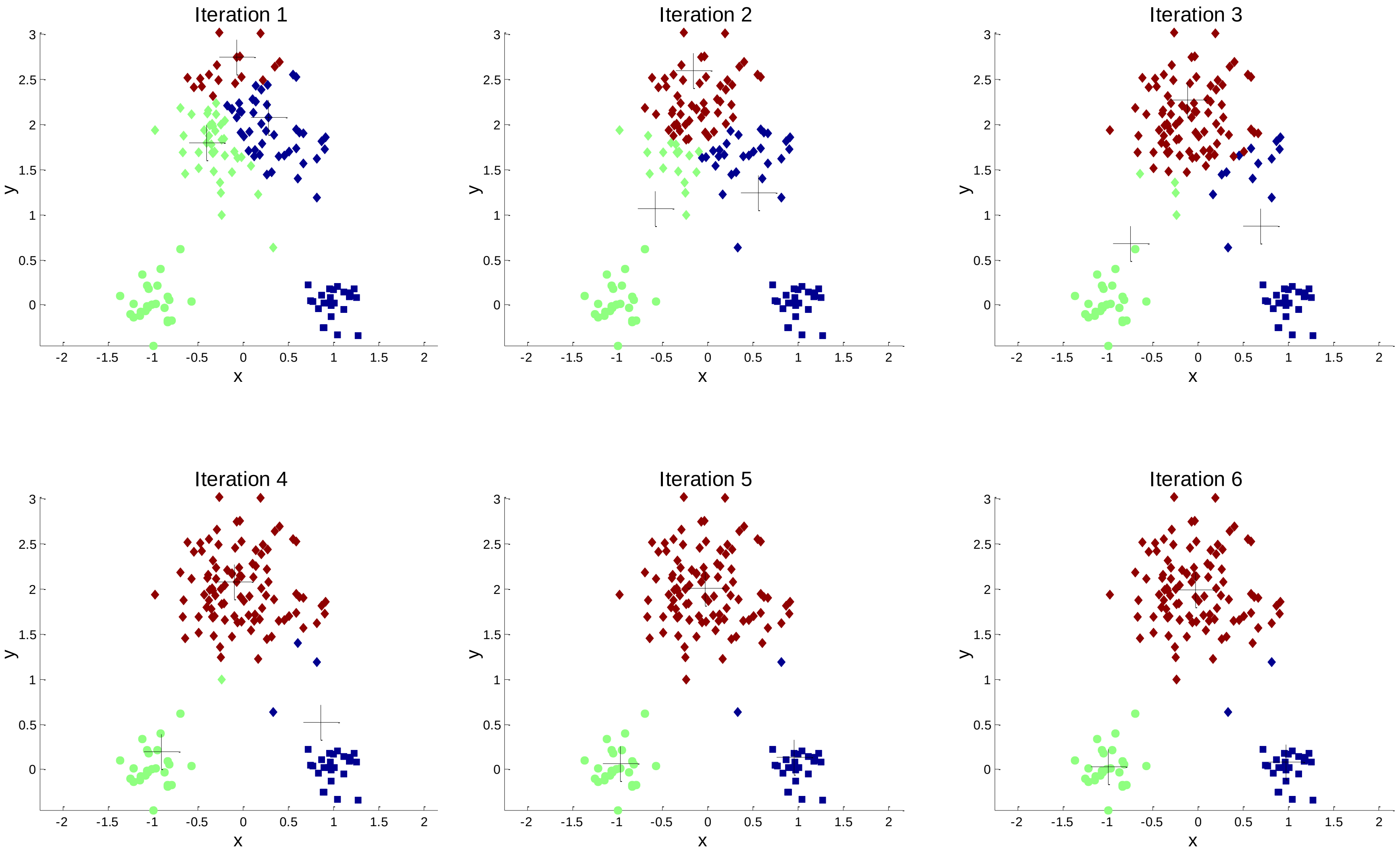
K-Means Example

- Given: {2,4,10,12,3,20,30,11,25}, k=2
- Randomly assign means: m_1=3,m_2=4
- K_1={2,3}, K_2={4,10,12,20,30,11,25}, m_1=2.5,m_2=16
- K_1={2,3,4},K_2={10,12,20,30,11,25}, m_1=3,m_2=18
- K_1={2,3,4,10},K_2={12,20,30,11,25}, m_1=4.75,m_2=19.6
- K_1={2,3,4,10,11,12},K_2={20,30,25}, m_1=7,m_2=25
- Stop as the clusters with these means are the same.

Example of K-means Clustering



Example of K-means Clustering



K-means Clustering – Details

- Simple iterative algorithm.
  - Choose initial centroids;
  - repeat {assign each point to a nearest centroid; re-compute cluster centroids}
  - until centroids stop changing.
- Initial centroids are often chosen randomly.
  - Clusters produced can vary from one run to another
- The centroid is (typically) the mean of the points in the cluster, but other definitions are possible.
- K-means will converge for common proximity measures with appropriately defined centroid.
- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to 'Until relatively few points change clusters'
- Complexity is O(n * K * I * d)
  - n = number of points, K = number of clusters,
    I = number of iterations, d = number of attributes

K-means Objective Function

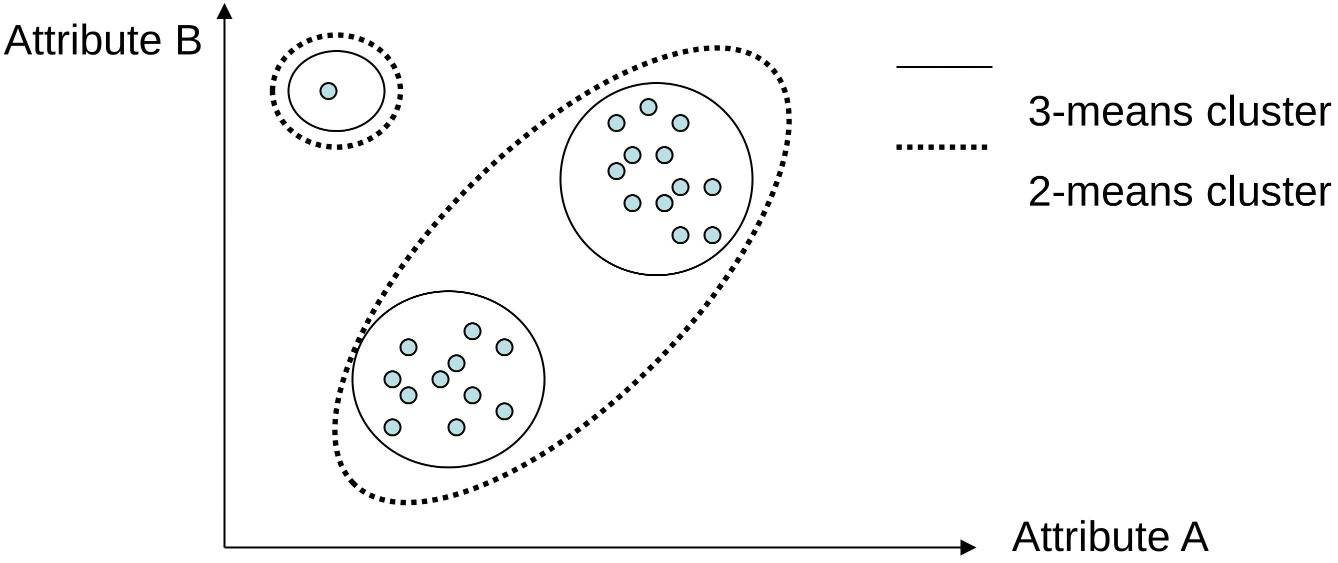
- A common objective function (used with Euclidean distance measure) is Sum of Squared Error (SSE)
  - For each point, the error is the distance to the nearest cluster center
  - To get SSE, we square these errors and sum them.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

  - *x* is a data point in cluster C_i and m_i is the centroid (mean) for cluster C_i
  - SSE improves in each iteration of K-means until it reaches a local or global minima.

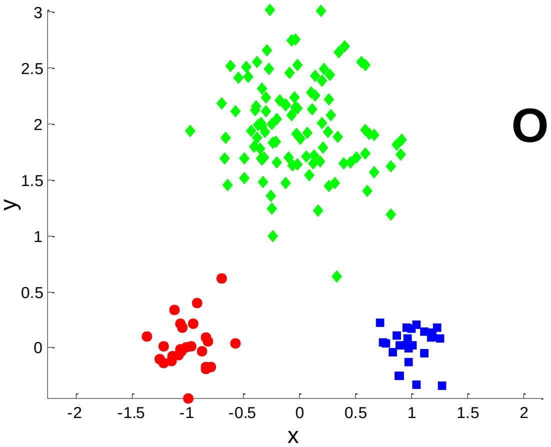
K-Means Clustering Problems

- Because k-means clustering techniques decide the number of clusters in advance, they are sensitive to **outliers**.
- Selection of initial centroids
- K-means works well for spherical clusters in small data sets, but poorly for complex-shaped clusters in large data sets.
- From the diagram below, you can see that, for this particular data set, using 3-means is better than using 2-means because the latter is excessively influenced by the outlier and fails to split the clusters well.
- Note that hierarchical, demographic, and neural clustering algorithms do not suffer from this drawback as they do not require you to specify in advance how many clusters to create.

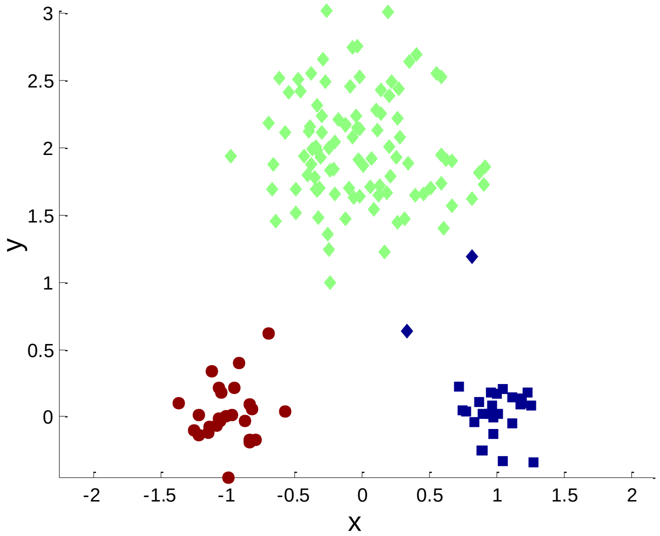


Source: Adapted from Dunham

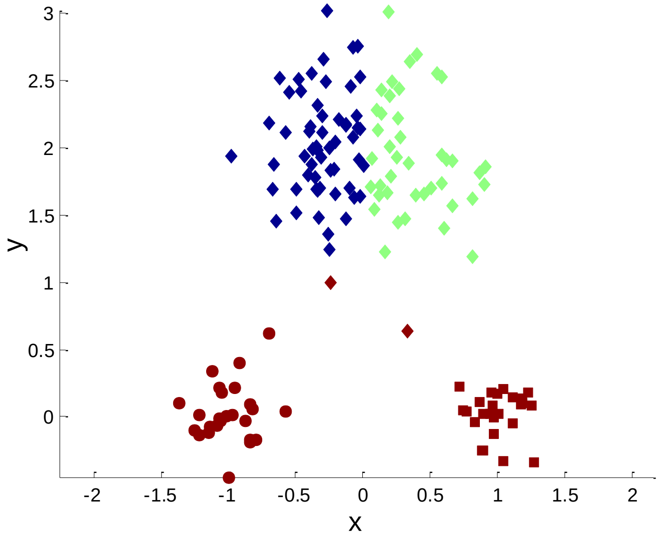
Two different K-means Clusterings



Original Points

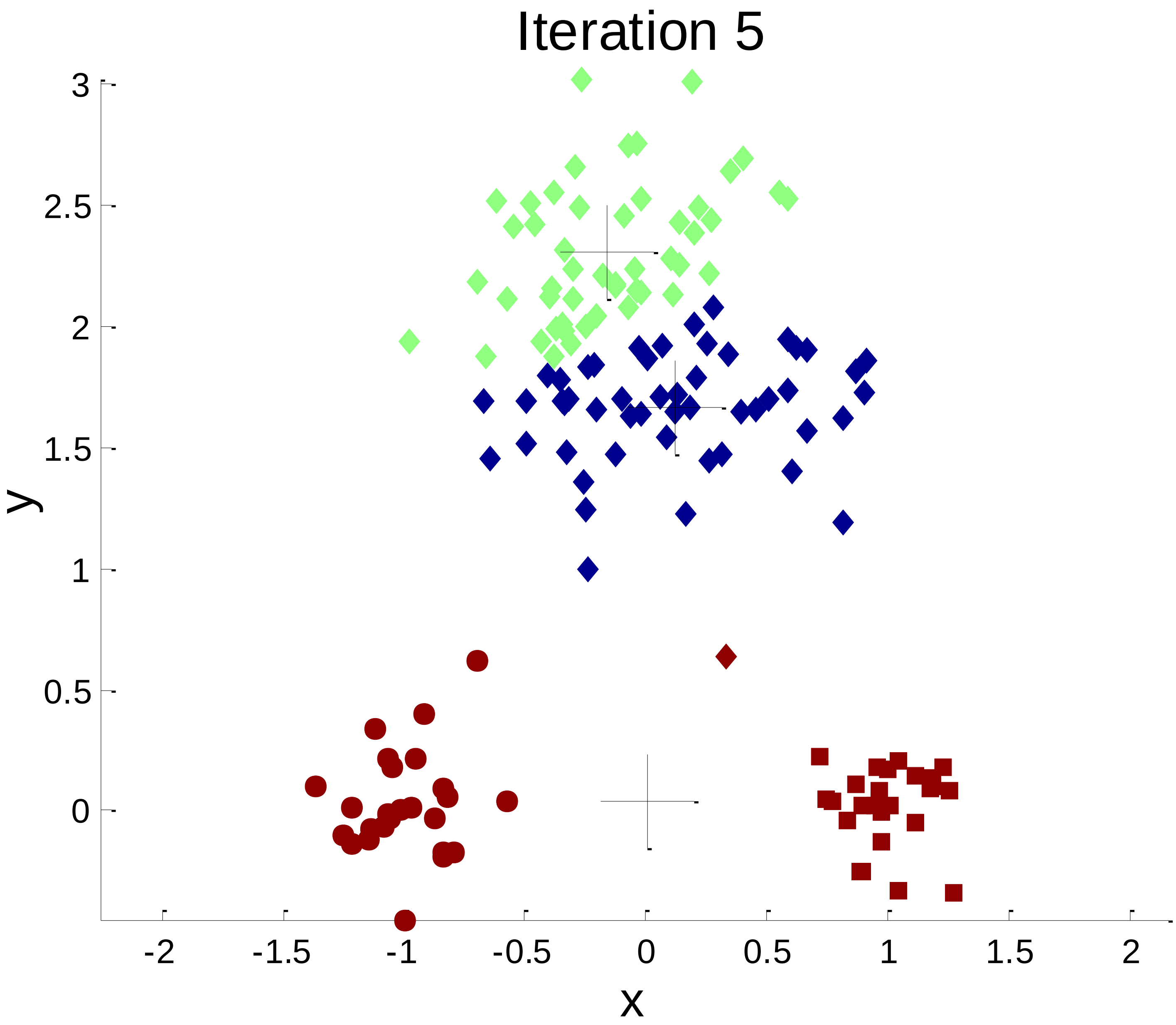


Optimal Clustering

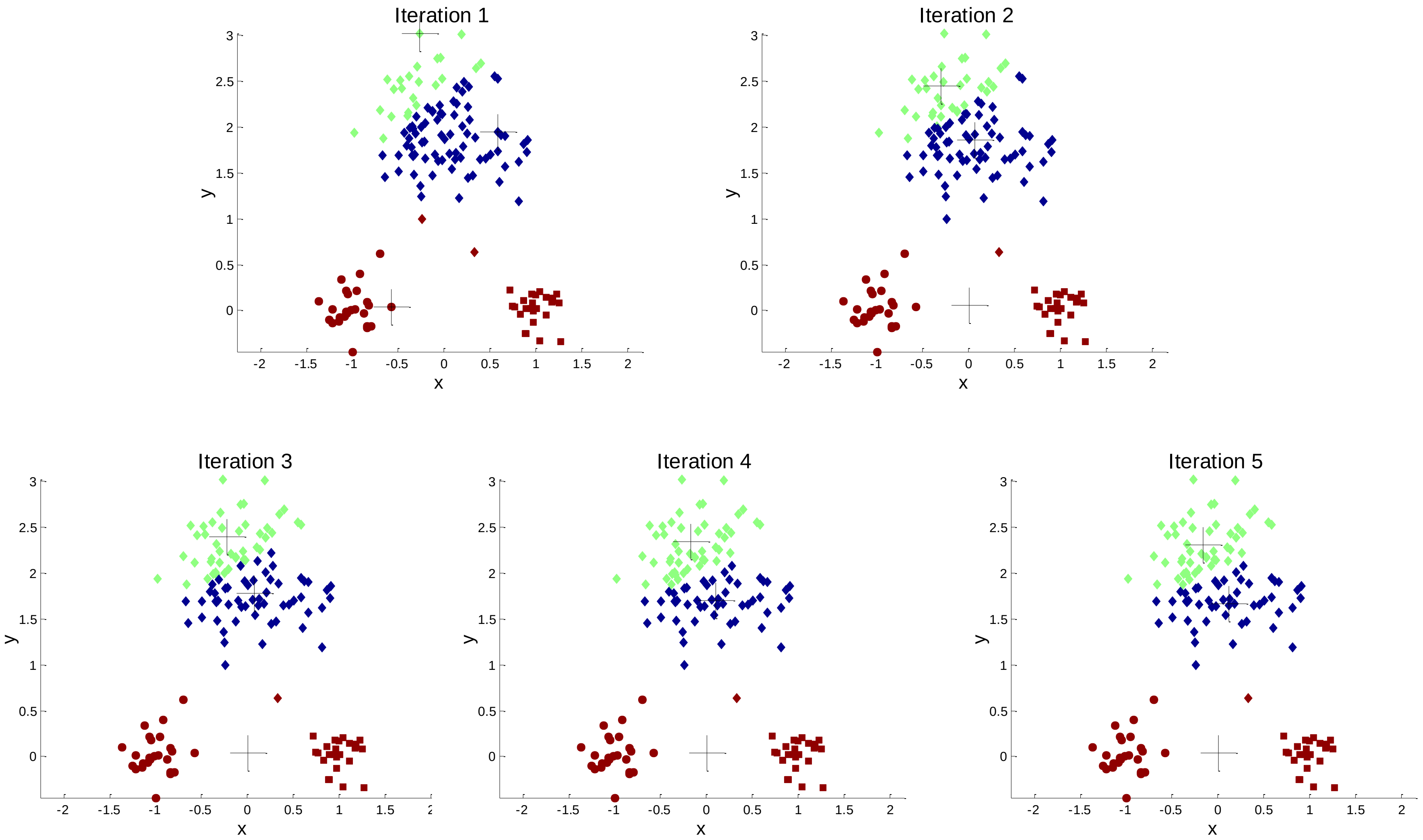


Sub-optimal Clustering

Importance of Choosing Initial Centroids …



Importance of Choosing Initial Centroids …


Iteration 1

Iteration 2

Iteration 3

Iteration 4

Iteration 5

Solutions to Initial Centroids Problem

- Multiple runs
  - Helps, but probability is not on your side
- Use some strategy to select the k initial centroids and then select among these initial centroids
  - Select most widely separated
    - K-means++ is a robust way of doing this selection
  - Use hierarchical clustering to determine initial centroids
- Bisecting K-means
  - Not as susceptible to initialization issues

K-means++

- This approach can be slower than random initialization, but very consistently produces better results in terms of SSE
  - The k-means++ algorithm guarantees an approximation ratio O(log k) in expectation, where k is the number of centers
- To select a set of initial centroids, *C*, perform the following

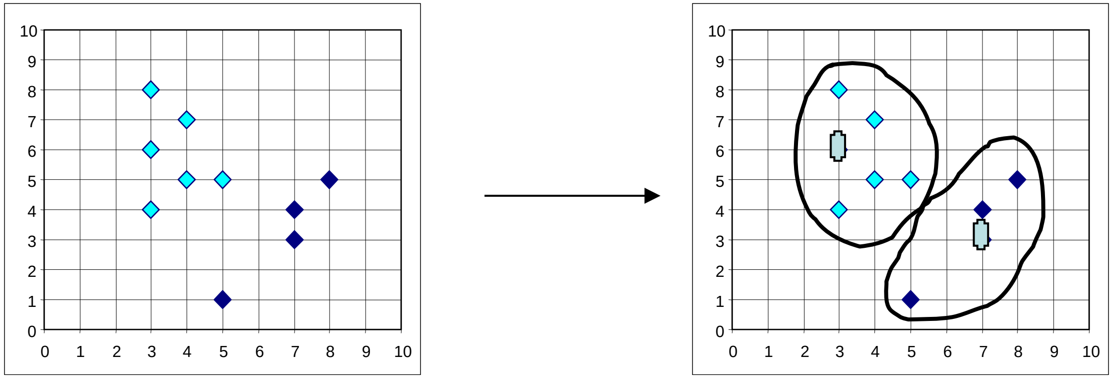
1. Select an initial point at random to be the first centroid
2. For k – 1 steps
3. For each of the N points, x_i, $1 \leq i \leq N$, find the minimum squared distance to the currently selected centroids, $C_1, \ldots, C_j$, $1 \leq j < k$, i.e., $\min_j d^2(C_j, x_i)$
4. Randomly select a new centroid by choosing a point with probability proportional to $\frac{\min_j d^2(C_j, x_i)}{\sum_i \min_j d^2(C_j, x_i)}$ is
5. End For

Other Variations of the *K-Means* Method

- A few variants of the *k-means* which differ in
  - Selection of the initial *k* means
  - Dissimilarity calculations
  - Strategies to calculate cluster means
- Handling categorical data: *k-modes* (Huang'98)
  - Replacing means of clusters with modes
  - Using new dissimilarity measures to deal with categorical objects
  - Using a frequency-based method to update modes of clusters
  - A mixture of categorical and numerical data: *k-prototype* method
- *K-Medoids (*Partitioning Around Medoids*) Clustering Method*
- *Bisecting K-Means*
- *CLARA* (Kaufmann & Rousseeuw, 1990)
- *CLARANS* (Ng & Han, 1994): Randomized sampling
- Focusing + spatial data structure (Ester et al., 1995)

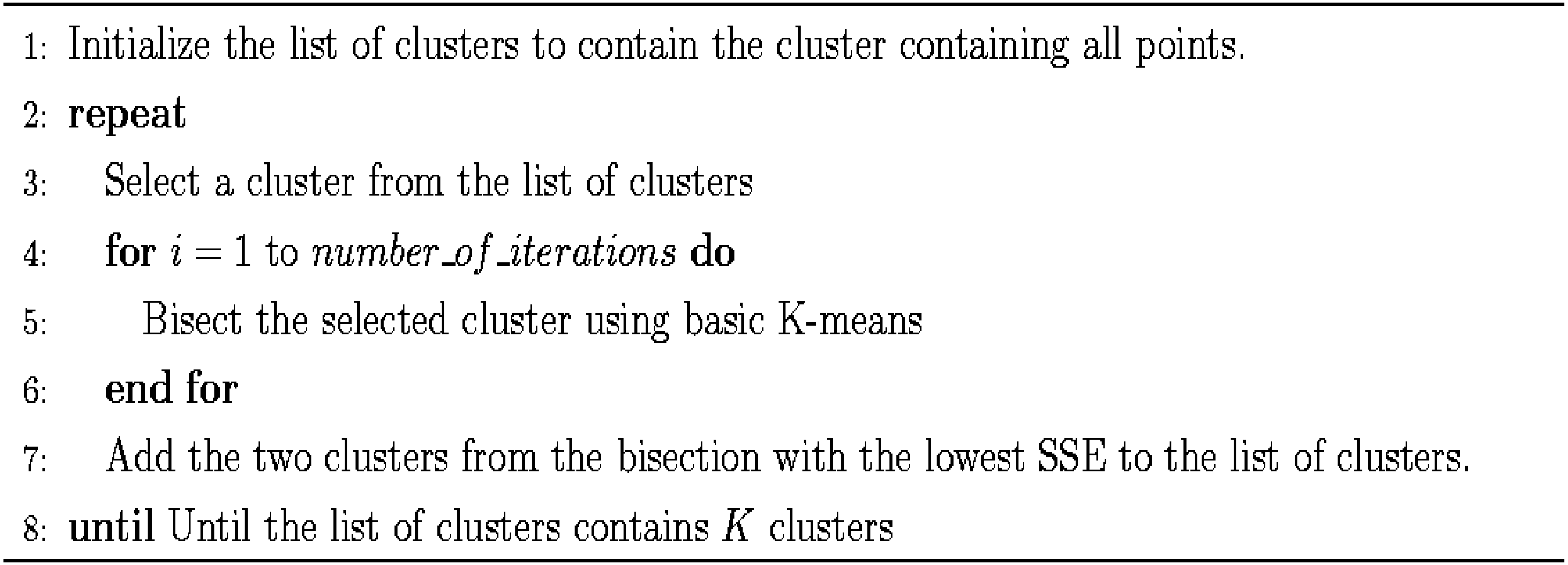
K-Medoids (Partitioning Around Medoids) Clustering Method

- The k-means algorithm is sensitive to outliers !
  - Since an object with an extremely large value may substantially distort the distribution of the data
- K-Medoids: Instead of taking the **mean** value of the object in a cluster as a reference point, **medoids** can be used, which is the **most centrally located** object in a cluster



Bisecting K-means

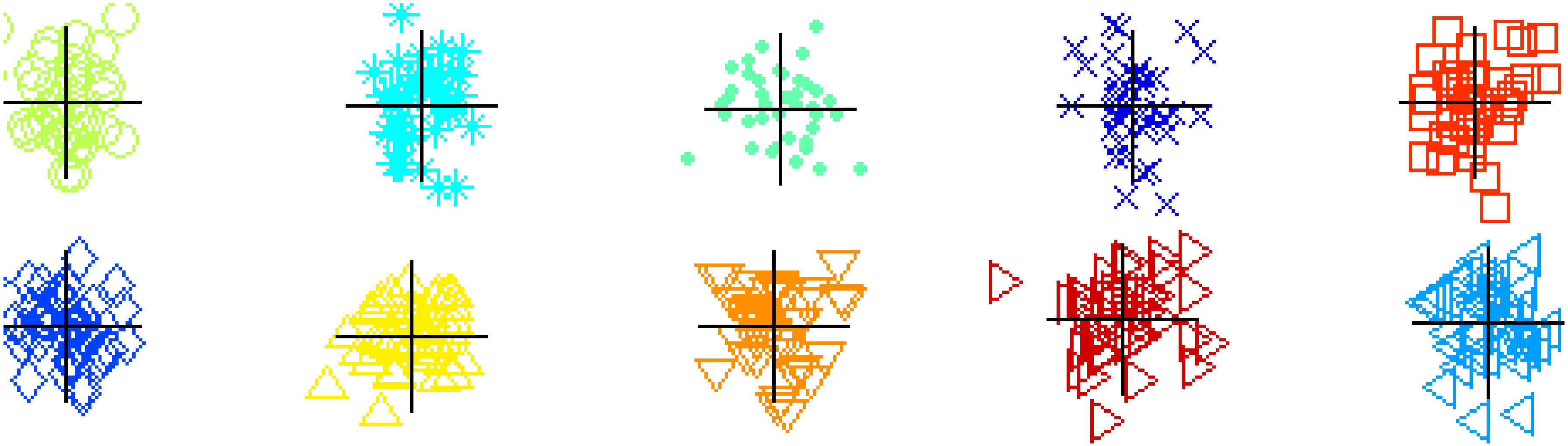
- Bisecting K-means algorithm
  - Variant of K-means that can produce a partitional or a hierarchical clustering



```
Initialize the list of clusters to contain the cluster containing all points.
repeat
    Select a cluster from the list of clusters
    for $i = 1$ to $number\_of\_iterations$ do
        Bisect the selected cluster using basic K-means
    end for
    Add the two clusters from the bisection with the lowest SSE to the list of clusters.
until Until the list of clusters contains $K$ clusters
```

CLUTO: http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview

Bisecting K-means Example



1.Initial Cluster: Start with all data points in one cluster.
2.Bisecting: Split the cluster into two sub-clusters using k-means (k=2).
3.Selection: Choose the cluster with the highest variance or largest size for further splitting.
4.Repeat: Continue bisecting until the desired number of clusters is reached.

In medicine, In medicine, bisecting k-means has been applied to tasks like brain tumor detection using MRI images

Limitations of K-means

- K-means has problems when clusters are of differing
  - Sizes
  - Densities
  - Non-globular shapes

- K-means has problems when the data contains outliers.
  - One possible solution is to remove outliers before clustering