Data Mining Cluster Analysis

General principles

&

K-means algorithm(s)

Main source: slides from "Lecture Notes for Chapter 7 -- Introduction to Data Mining, 2nd Edition", by Tan, Steinbach, Karpatne, Kumar

Preliminaries

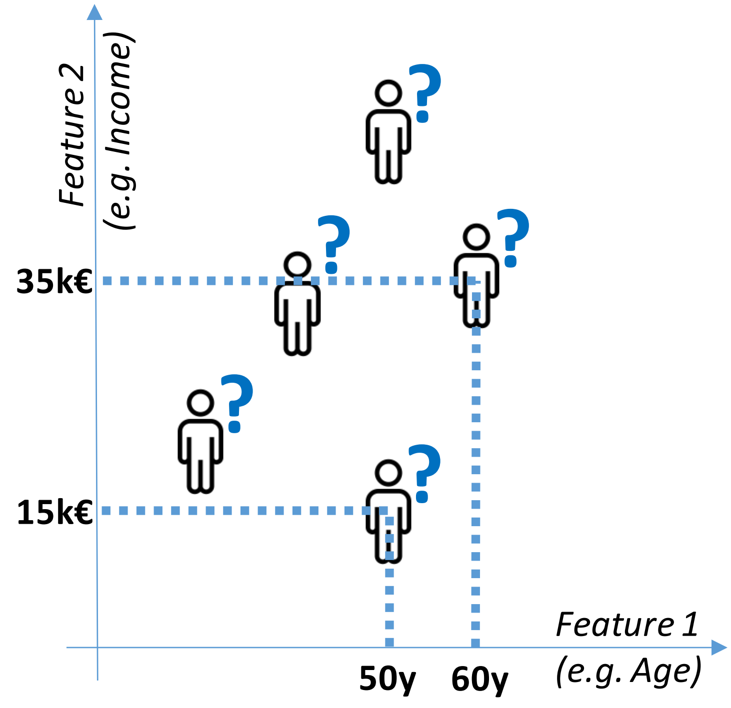
Classification and Clustering

Two members of the same family

- **Classification**
  - Aims to learn how to recognize objects of different categories
  - Categories are pre-defined: good vs bad, etc.
- **Clustering**
  - Aims to understand if the data seem to form different categories
  - Categories are not defined in advance

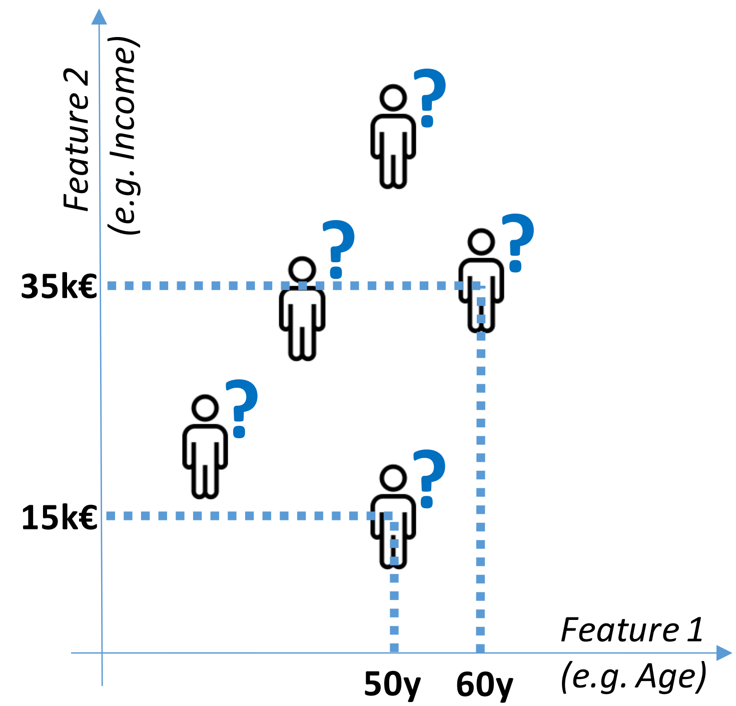
A common basic setting

- What we have
  - A set of objects, each of them described by some features
    - people described by age, gender, height, etc.
    - bank transactions described by type, amount, time, etc.



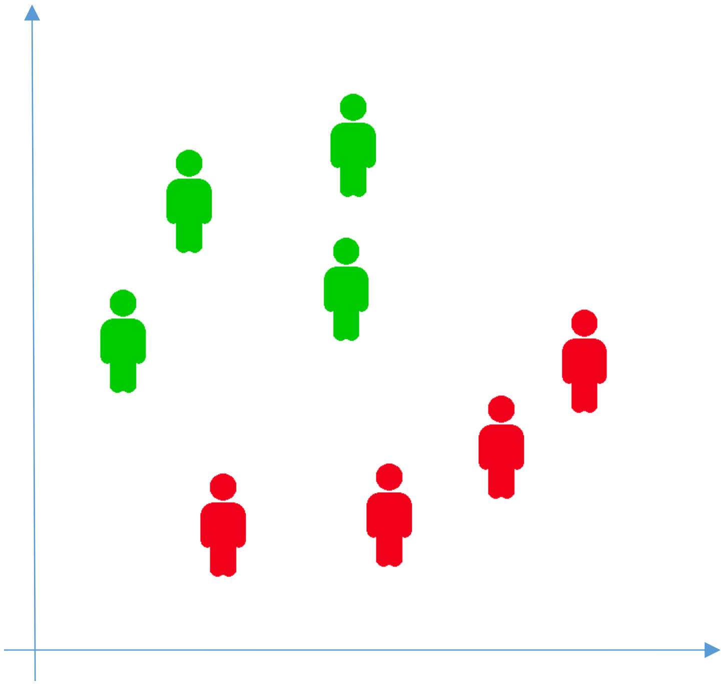
Classification problem

- What we want to do
  - Associate the objects of a set to a class, taken from a predefined list
    - "good customer" vs. "churner"
    - "normal transaction" vs. "fraudulent"
    - "low risk patient" vs. "risky"



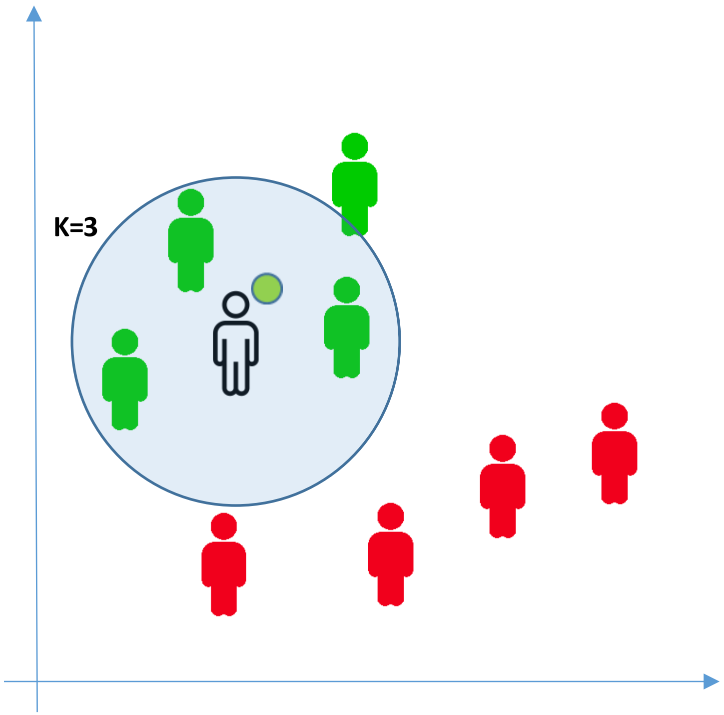
Classification problem

- What we know
  - No domain knowledge or theory
  - Only examples: Training Set
    - Subset of labelled objects
- What we can do
  - Learn from examples
  - Make inferences about the other objects



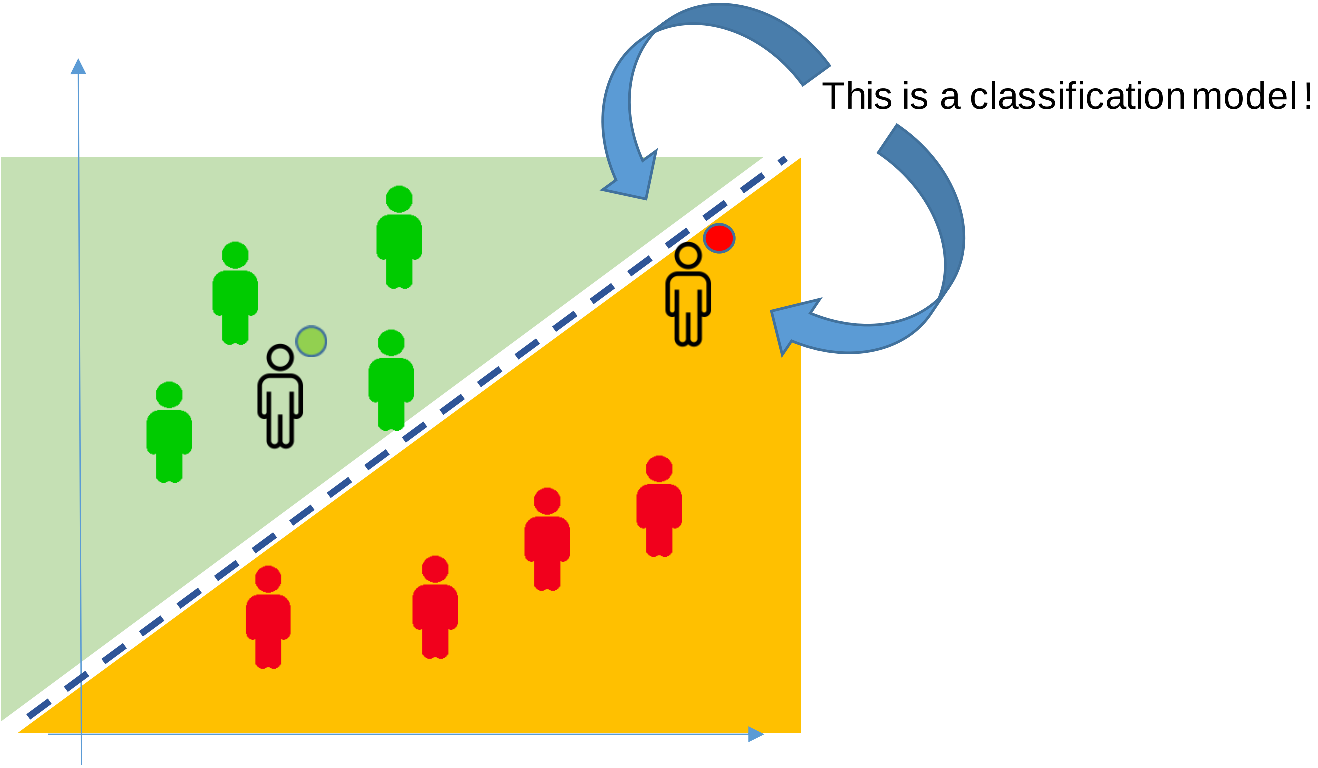
Classify by similarity

- K-Nearest Neighbors
  - Decide label based on K most similar examples



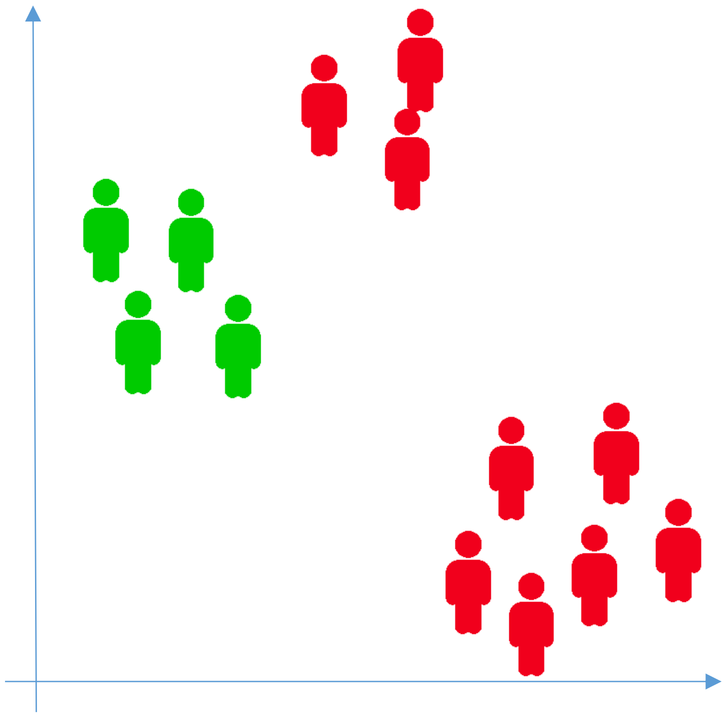
Build a model

- Basic example: linear separation line



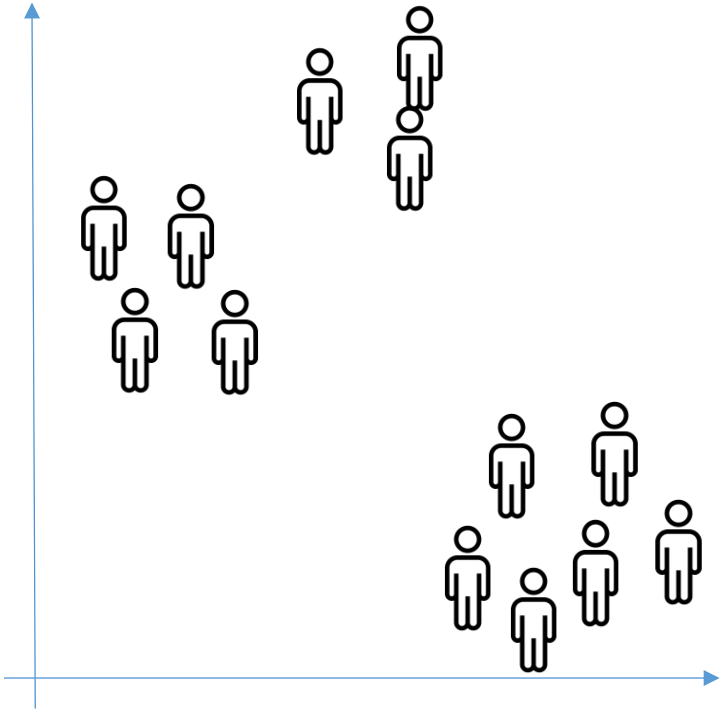
Clustering

- Classification starts from predefined labels and some examples to learn



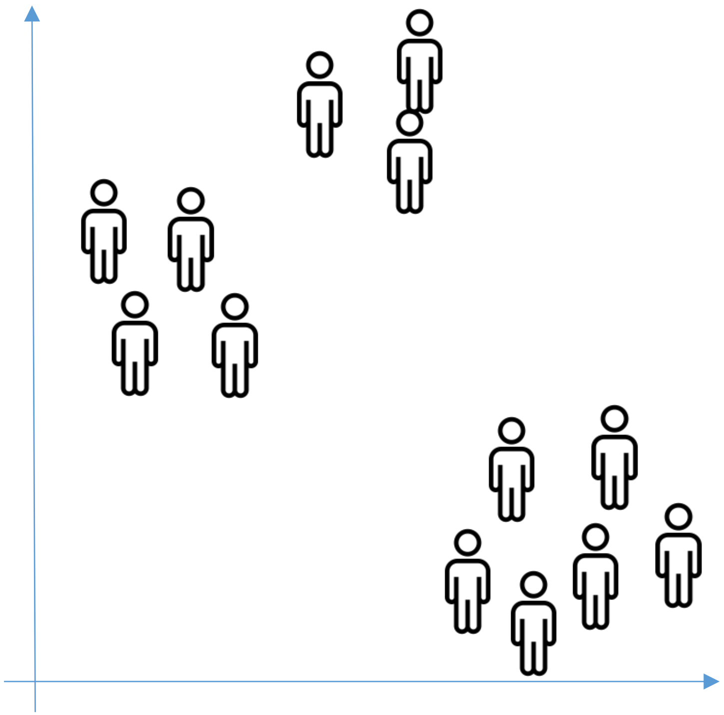
Clustering

- What if no labels are known?
  - We might lack examples
  - Labels might actually not exist at all…



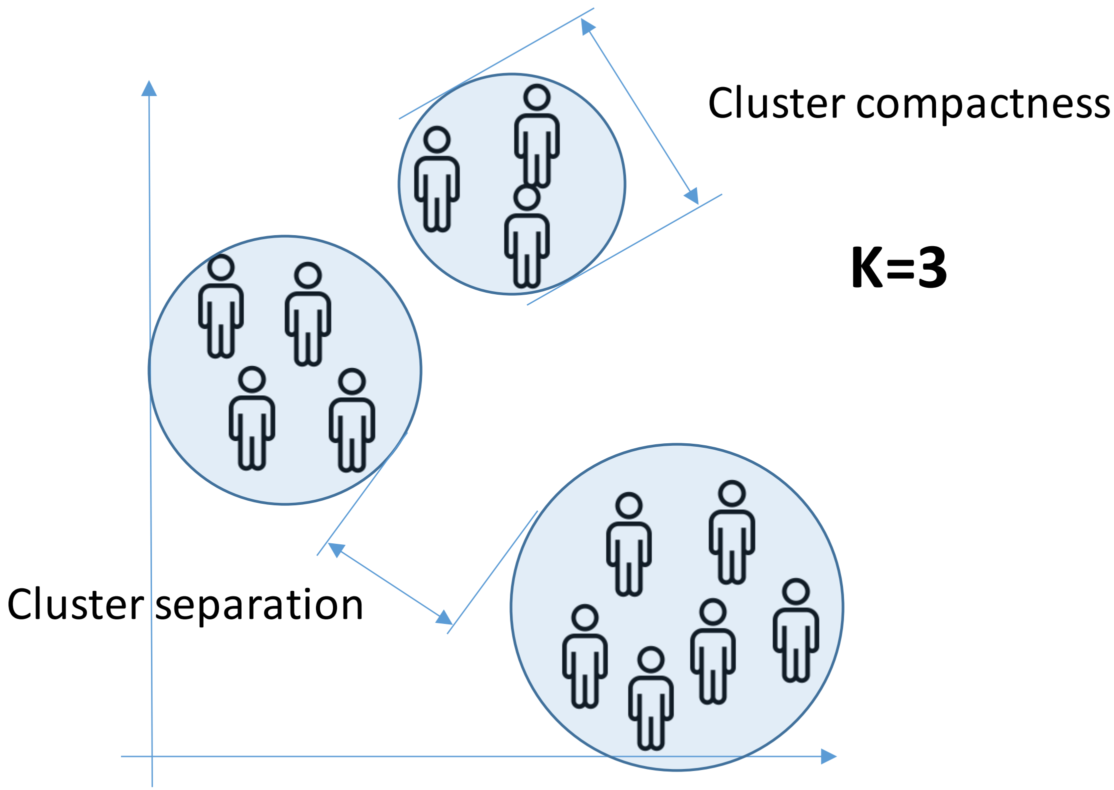
Clustering

- Objective: find structure in the data
- Group objects into clusters of "similar" entities



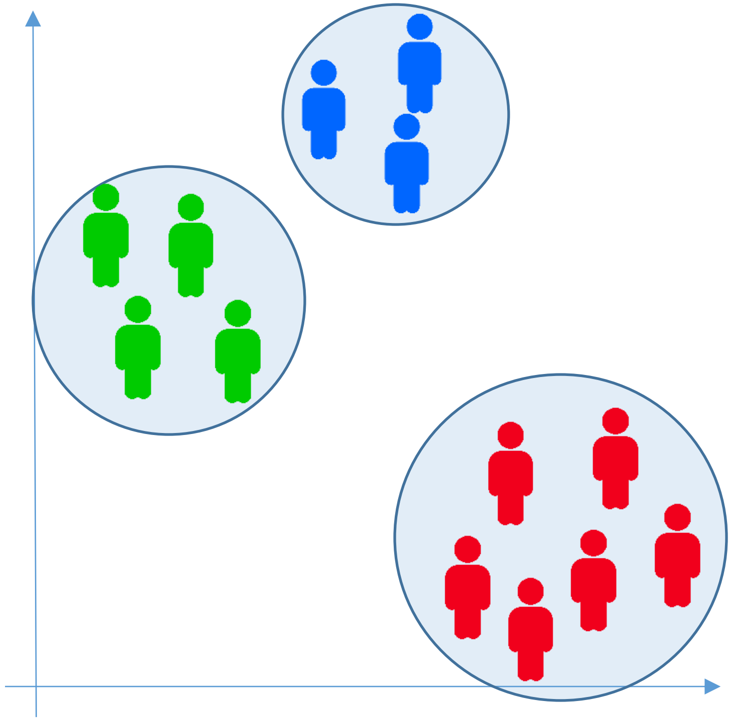
Clustering: K-means (sneak peek)

- Find k subgroups that form compact and well-separated clusters



Clustering: K-means (sneak peek)

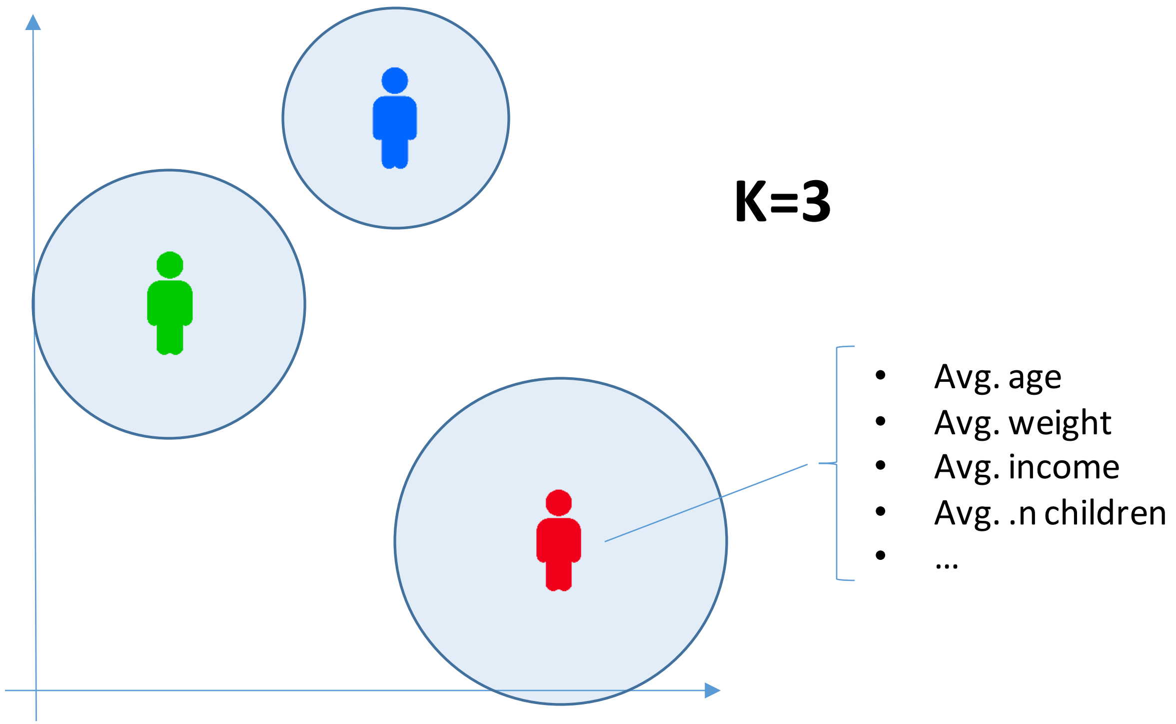
- Output 1: a partitioning of the initial set of objects



K=3

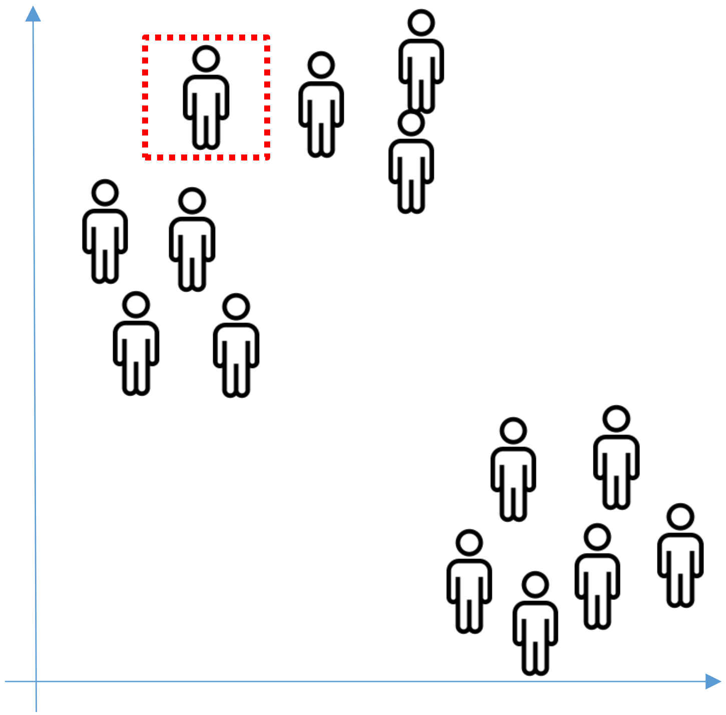
Clustering: K-means (sneak peek)

- Output 2: K representative objects (centroids)
- Centroid = average profile of the objects in the cluster



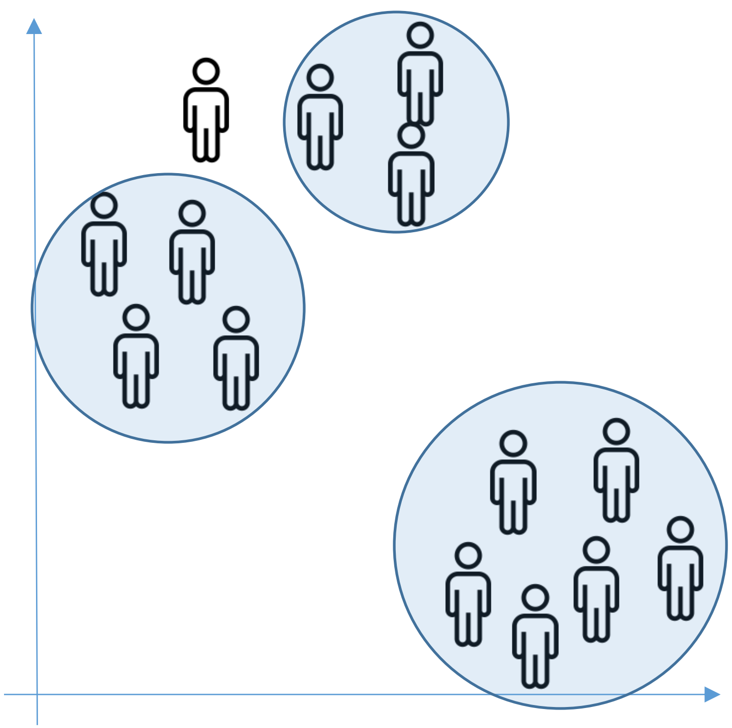
Clustering

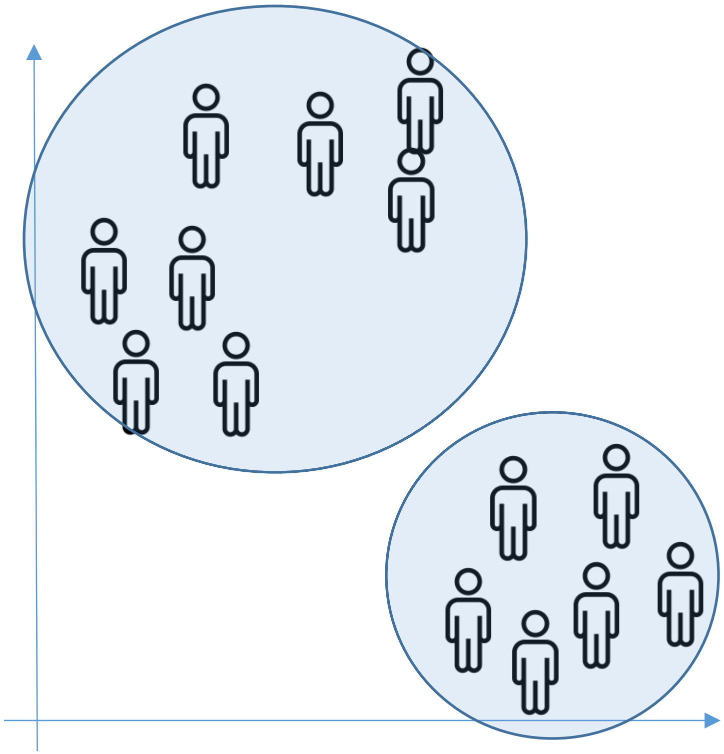
- What if we add one element?



Clustering

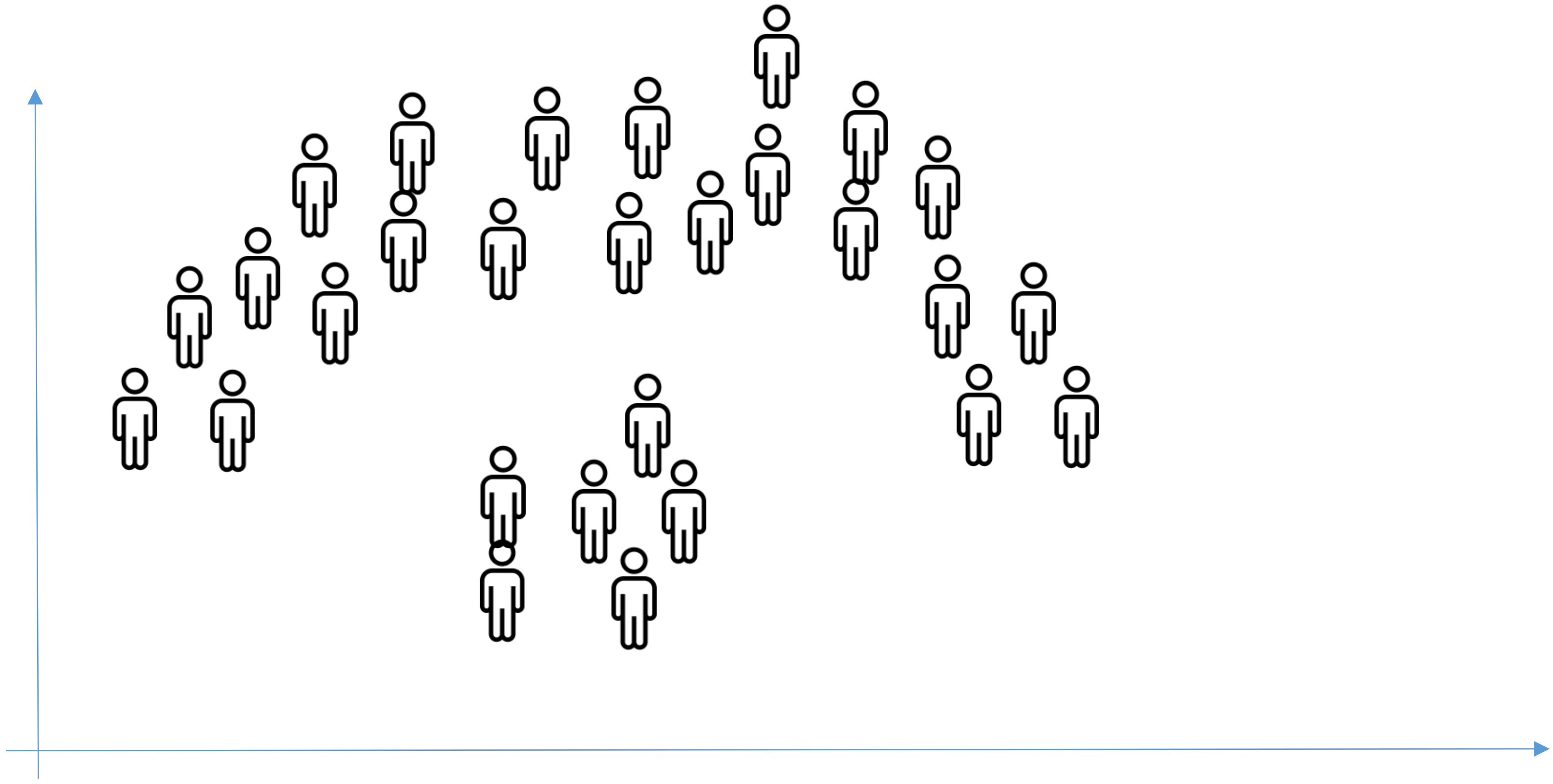
- What if we add one element?
- Which clustering do you like most?





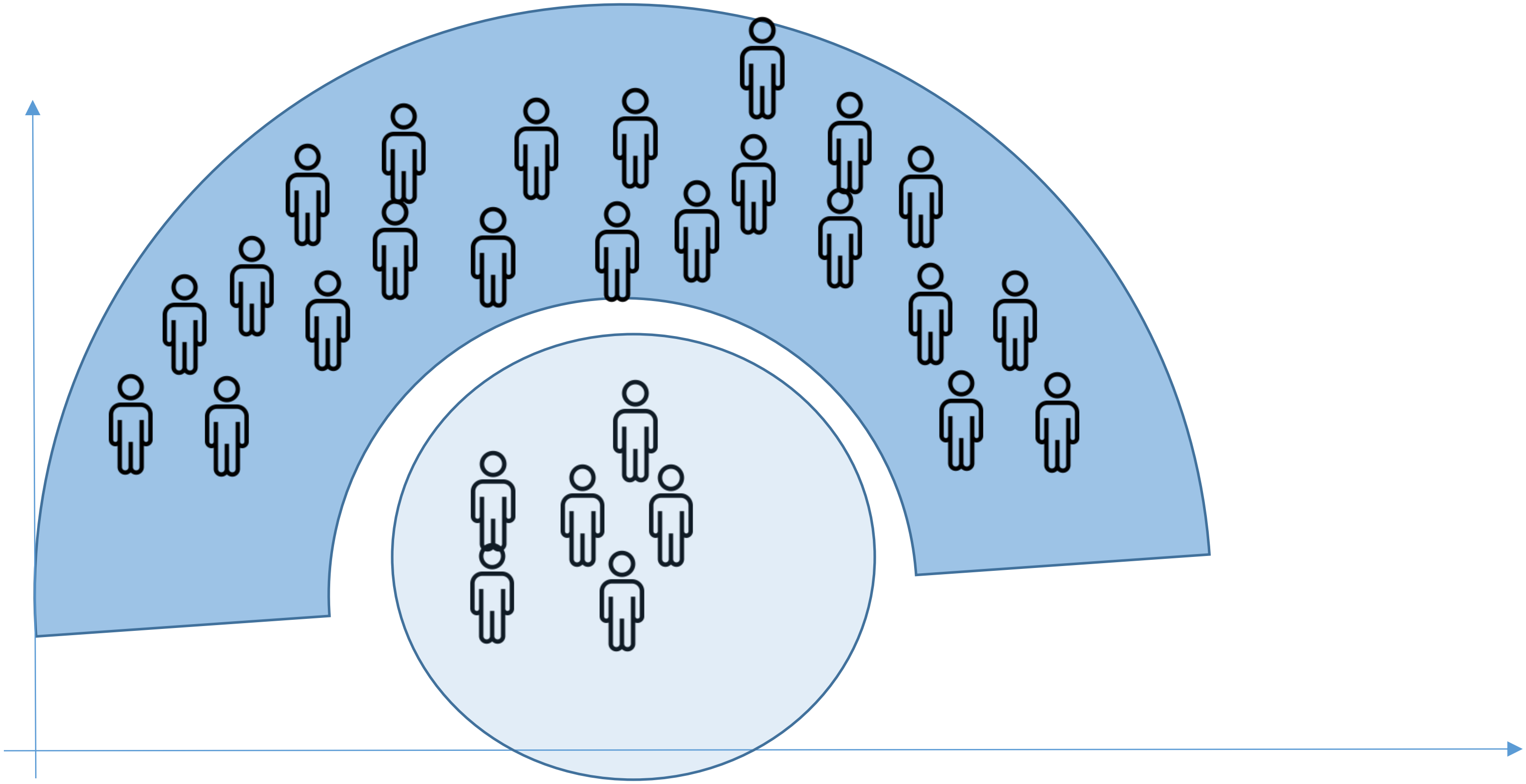
Clustering

- What about this ?!?



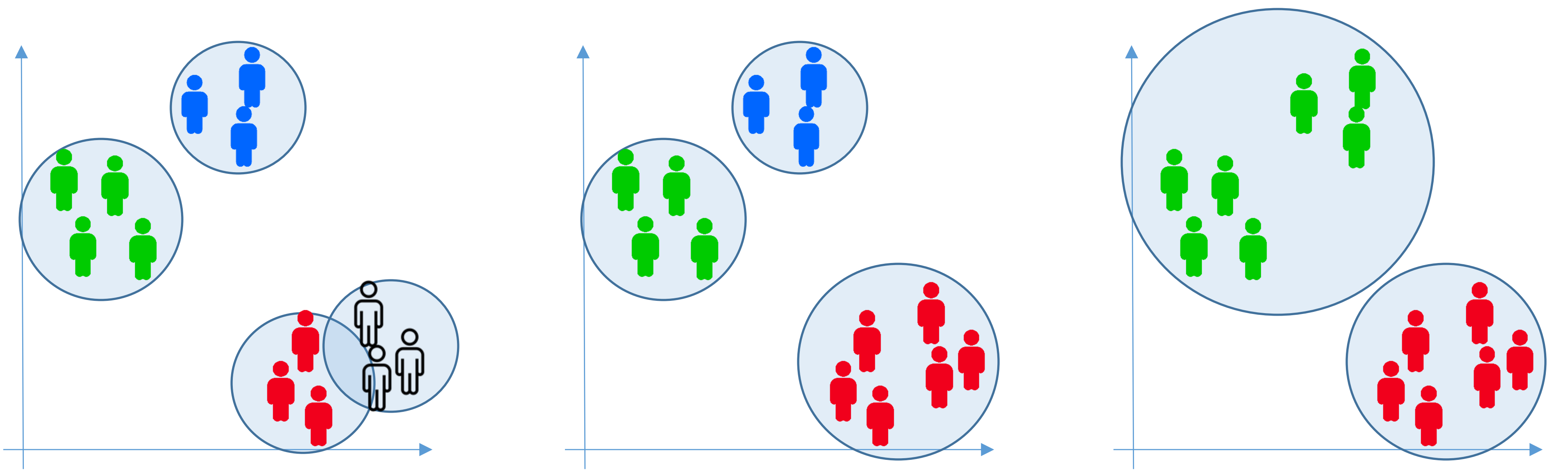
Clustering

- What about this ?!?



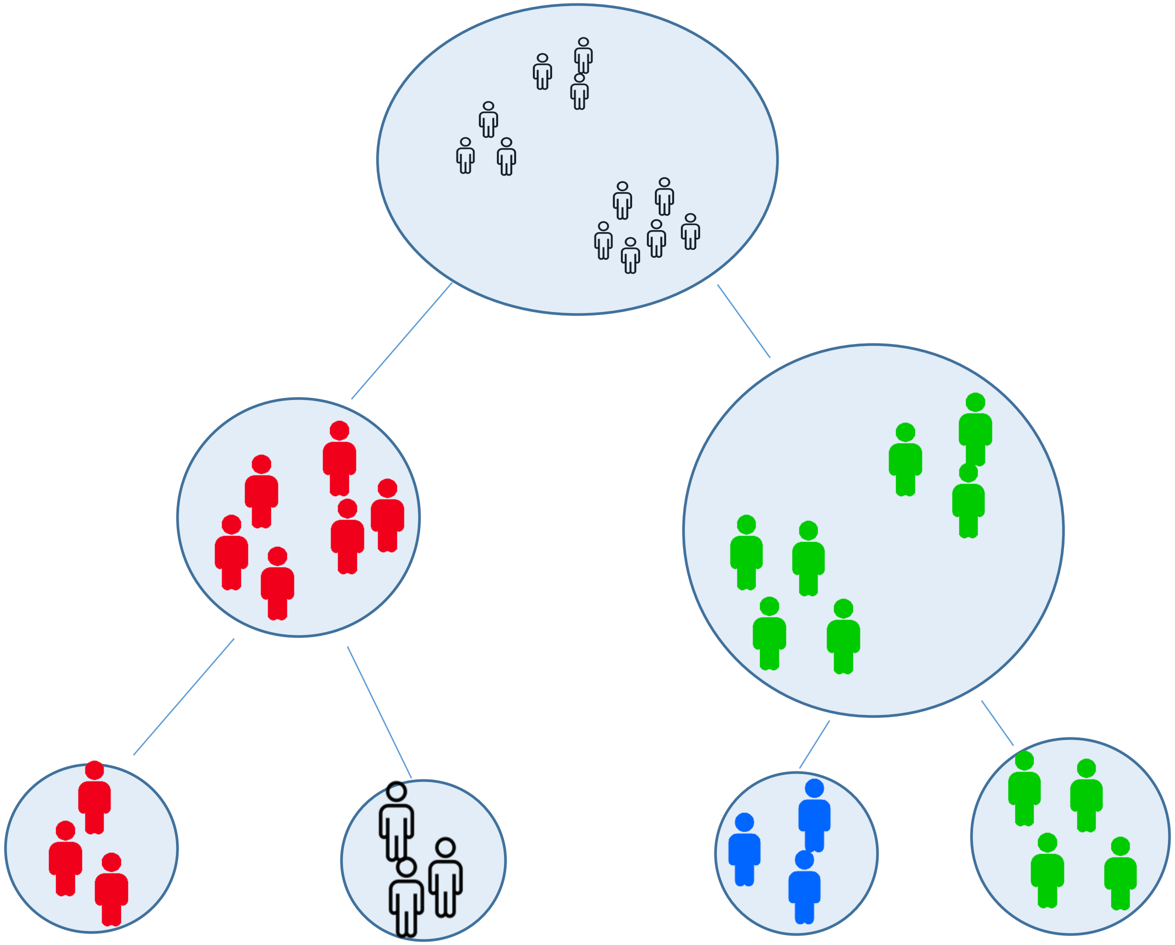
Clustering: hierarchical structures

- Sometimes we could have (or desire) multiple levels of aggregation



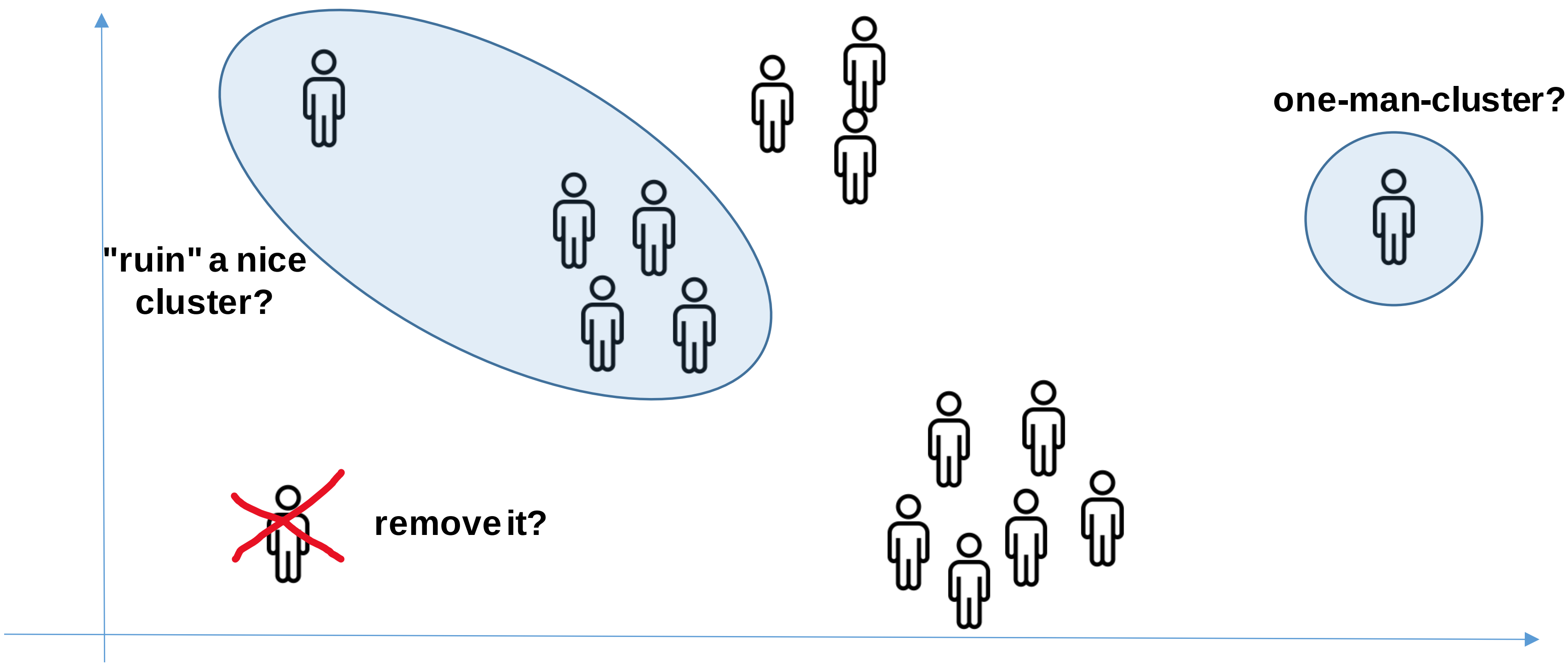
Clustering: hierarchical structures

- A hierarchy of clusters



Clustering everything ?

- What do we do with Outliers?

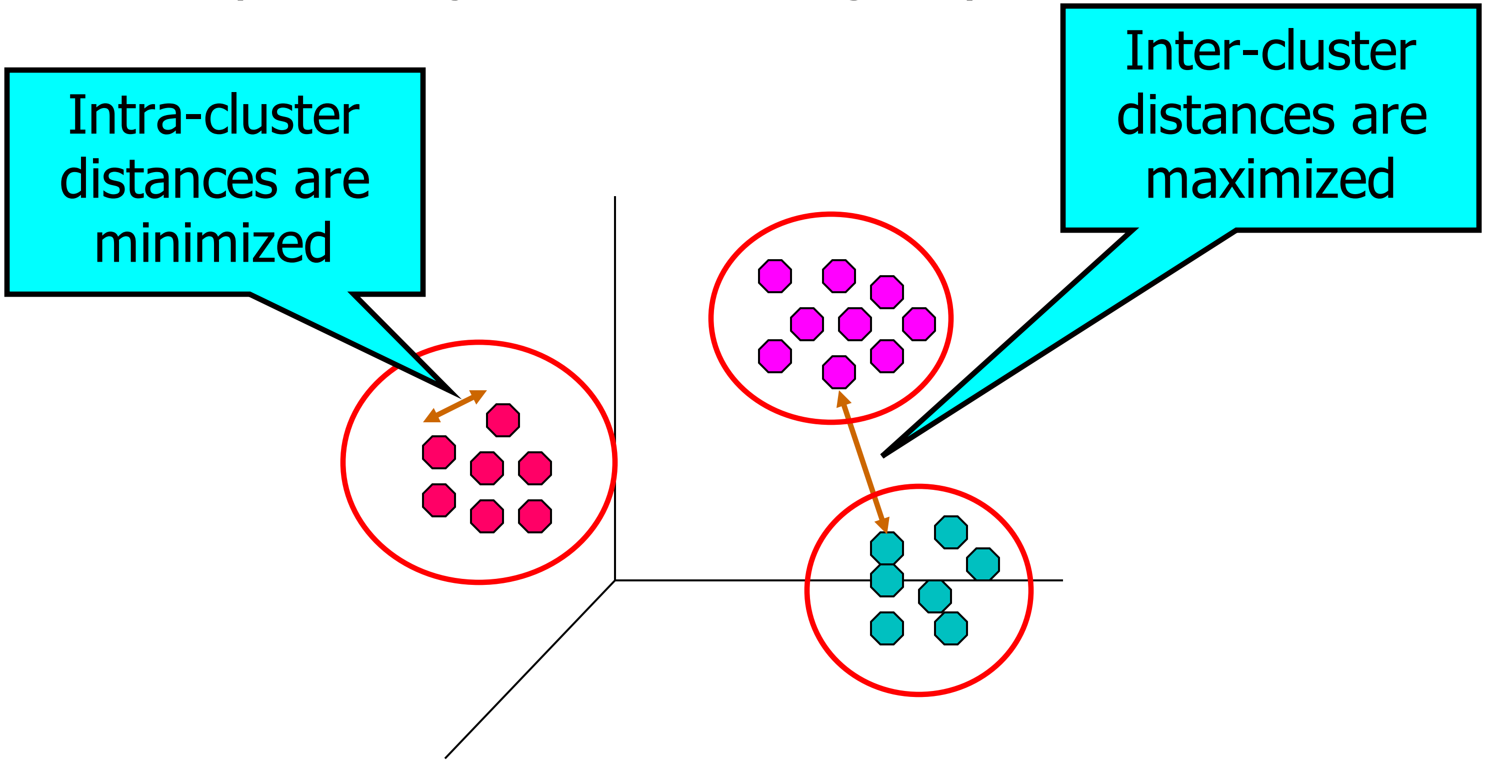


Basics

The many notions of «cluster»

What is Cluster Analysis?

- **Traditional definition (but not universal!):** finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups

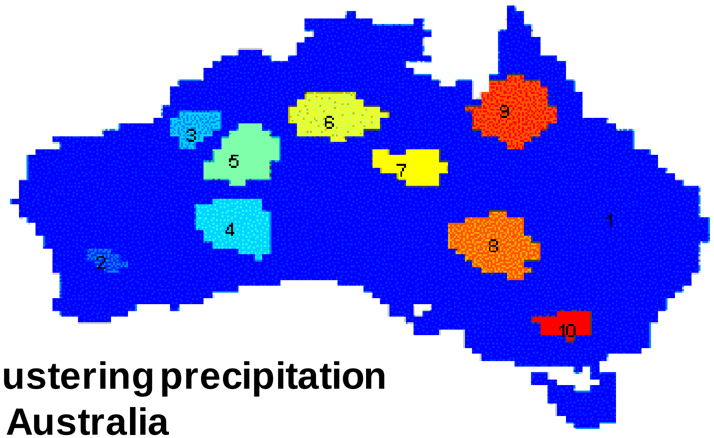


Applications of Cluster Analysis

- **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

What is not Cluster Analysis?

- Simple segmentation
  - Dividing students into different registration groups alphabetically, by last name

- Results of a query
  - Groupings are a result of an external specification
  - Clustering is a grouping of objects based on the data

- Supervised classification
  - Have class label information

- Association Analysis
  - based on groups, but different meaning

  --> will be a topic of (much) later classes

Notion of a Cluster can be Ambiguous

How many clusters?

Six Clusters

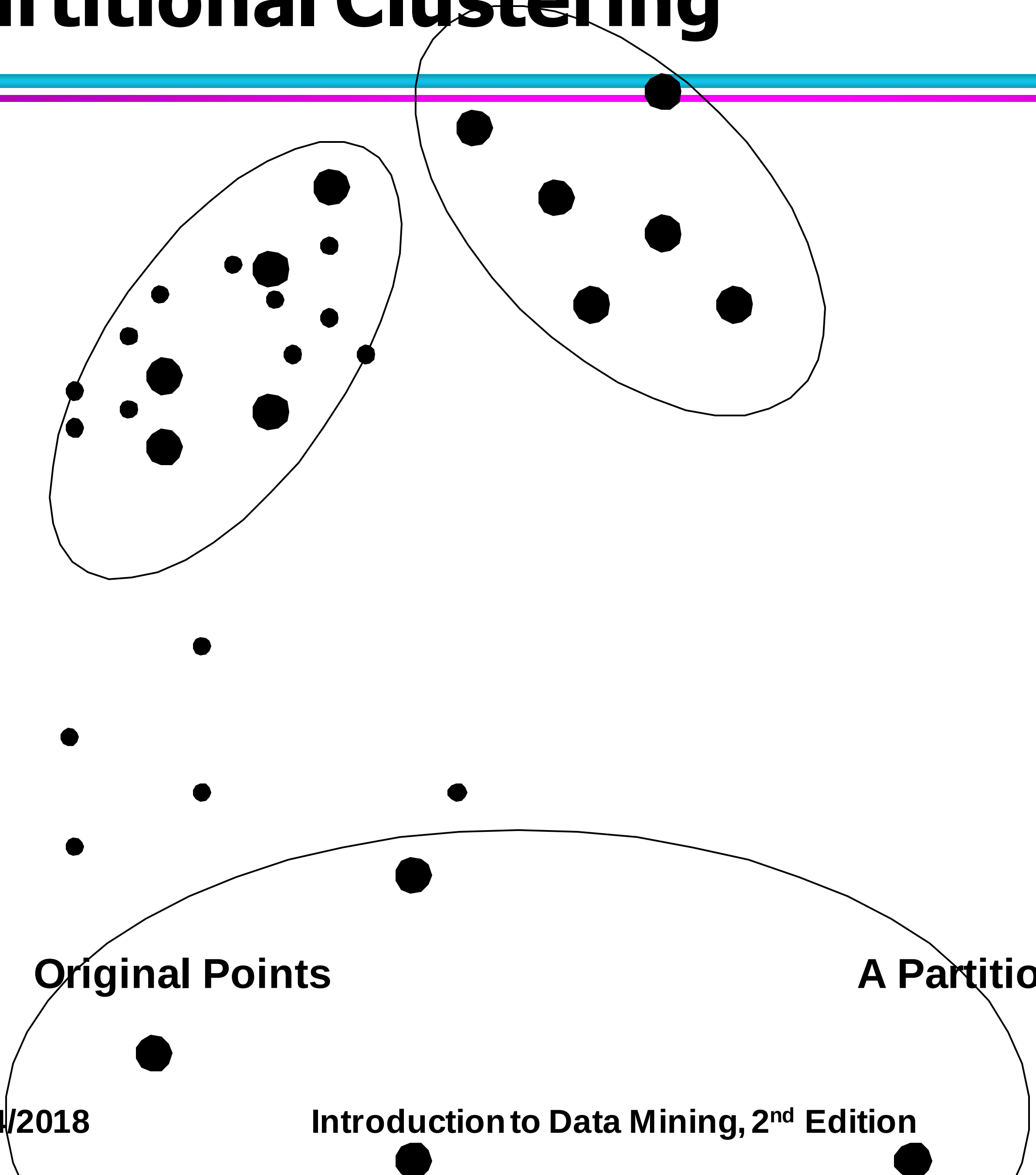
Two Clusters

Four Clusters

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) such that each data object is in exactly one subset
- Hierarchical clustering
  - A set of nested clusters organized as a hierarchical tree

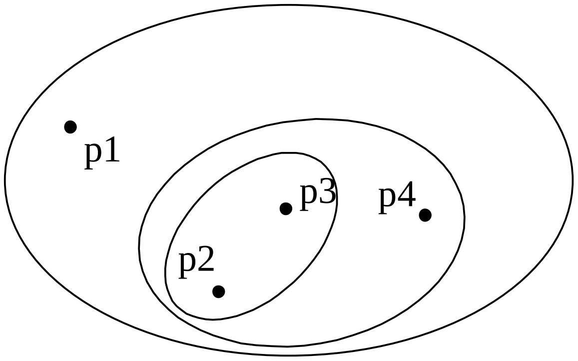
Partitional Clustering



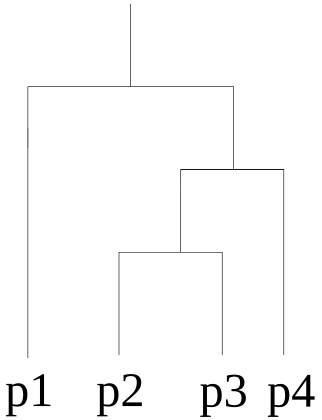
Original Points

A Partitional Clustering

Hierarchical Clustering



Traditional Hierarchical Clustering



Traditional Dendrogram

Other Distinctions Between Sets of Clusters

- Exclusive versus non-exclusive
  - In non-exclusive clusterings, **points may belong to multiple clusters.**
  - Can represent multiple classes or ‘**border**’ points
- Fuzzy versus non-fuzzy
  - 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

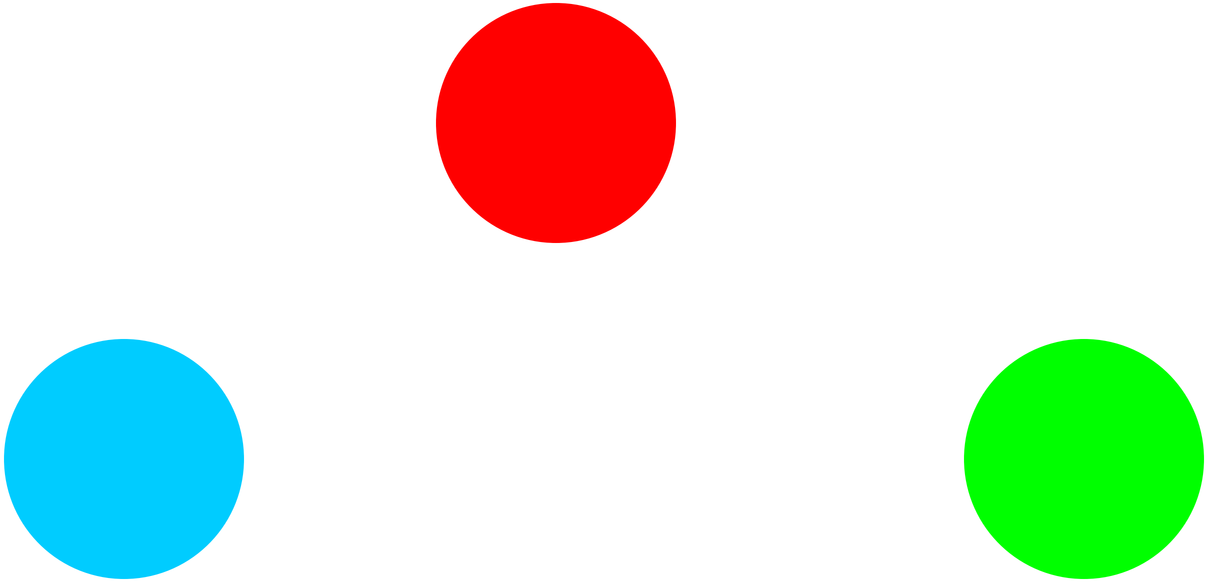
Types of Clusters

Or, possible features that clusters are expected to show:

- Well-separated clusters
- Center-based clusters
- Contiguous clusters
- Density-based clusters
- Property or concept 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

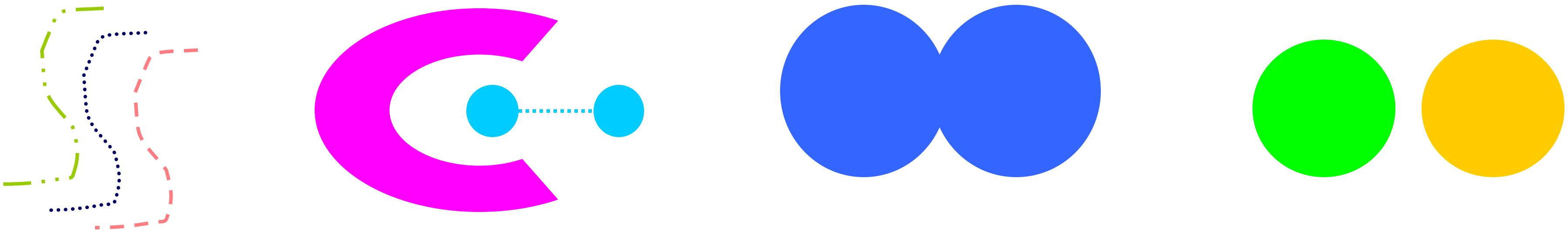
Types of Clusters: Center-Based

- Center-based
  - A cluster is a set of objects such that an object in a cluster is closer (more similar) to the "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)
  - Each point is closer to at least one point in its cluster than to any point in another cluster.
  - Graph based clustering

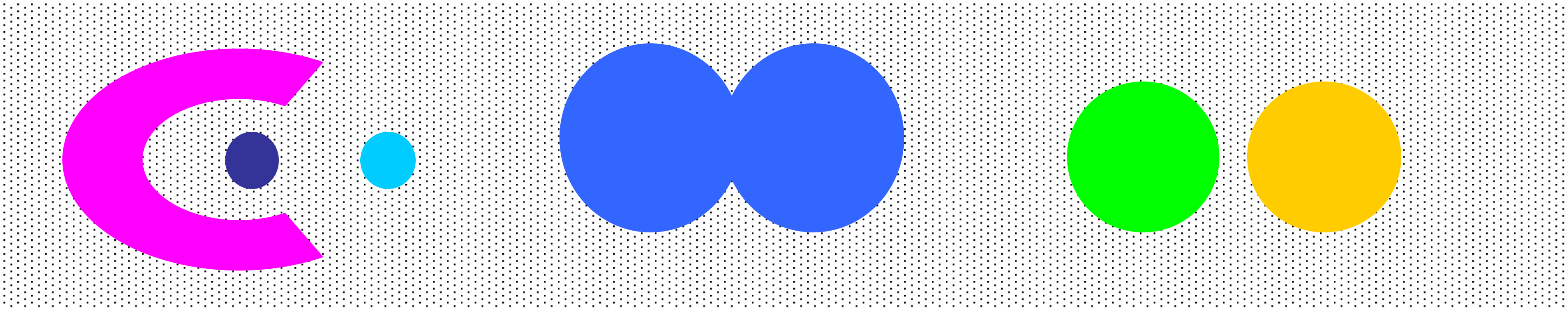


- This approach can **have trouble when noise is present** since a small bridge of points can **merge two distinct clusters**

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

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

Methods

Three fundamental clustering algorithms

Clustering Algorithms

- K-means and its variants
- Hierarchical clustering
- Density-based clustering

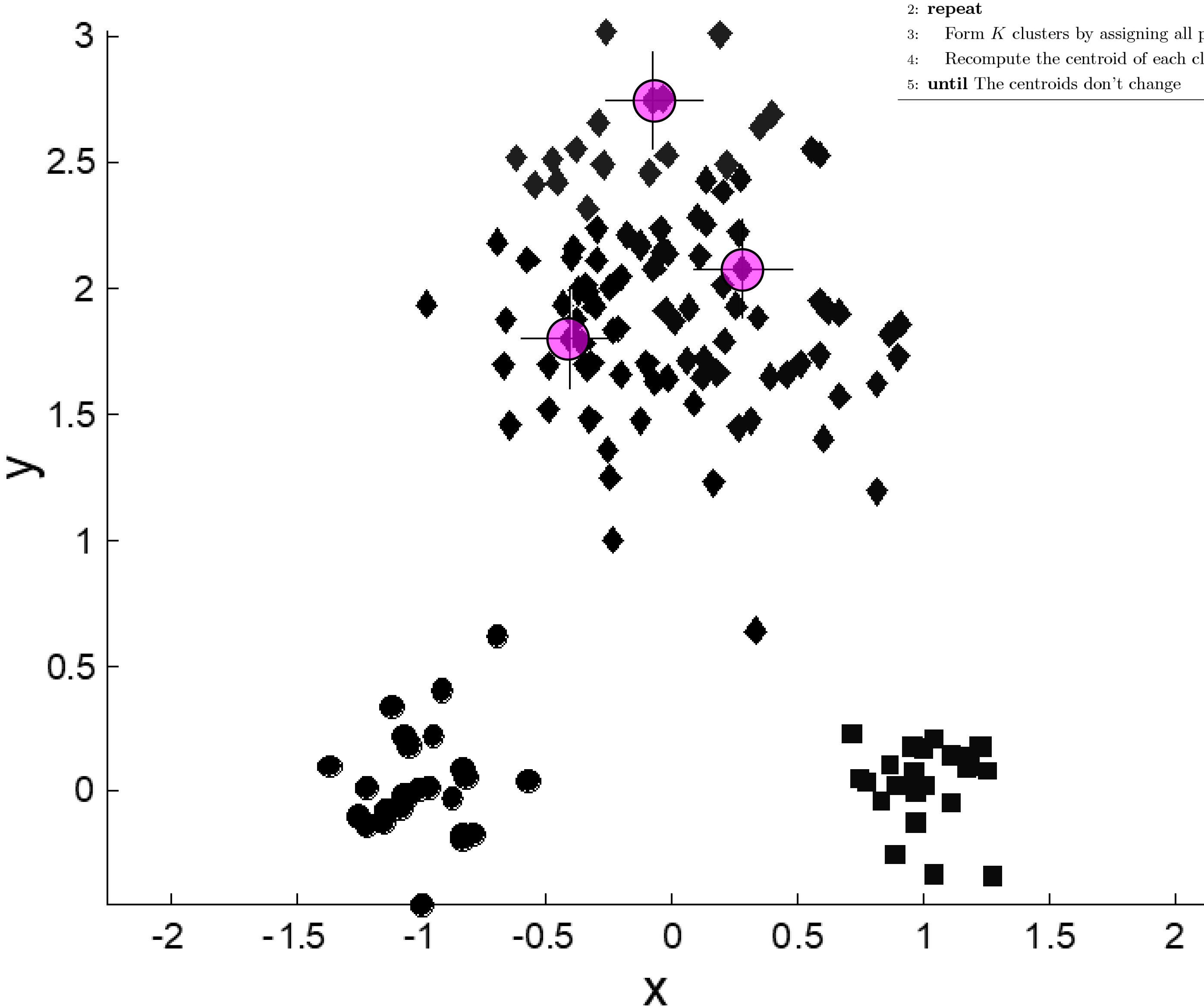
K-means Clustering

- 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:

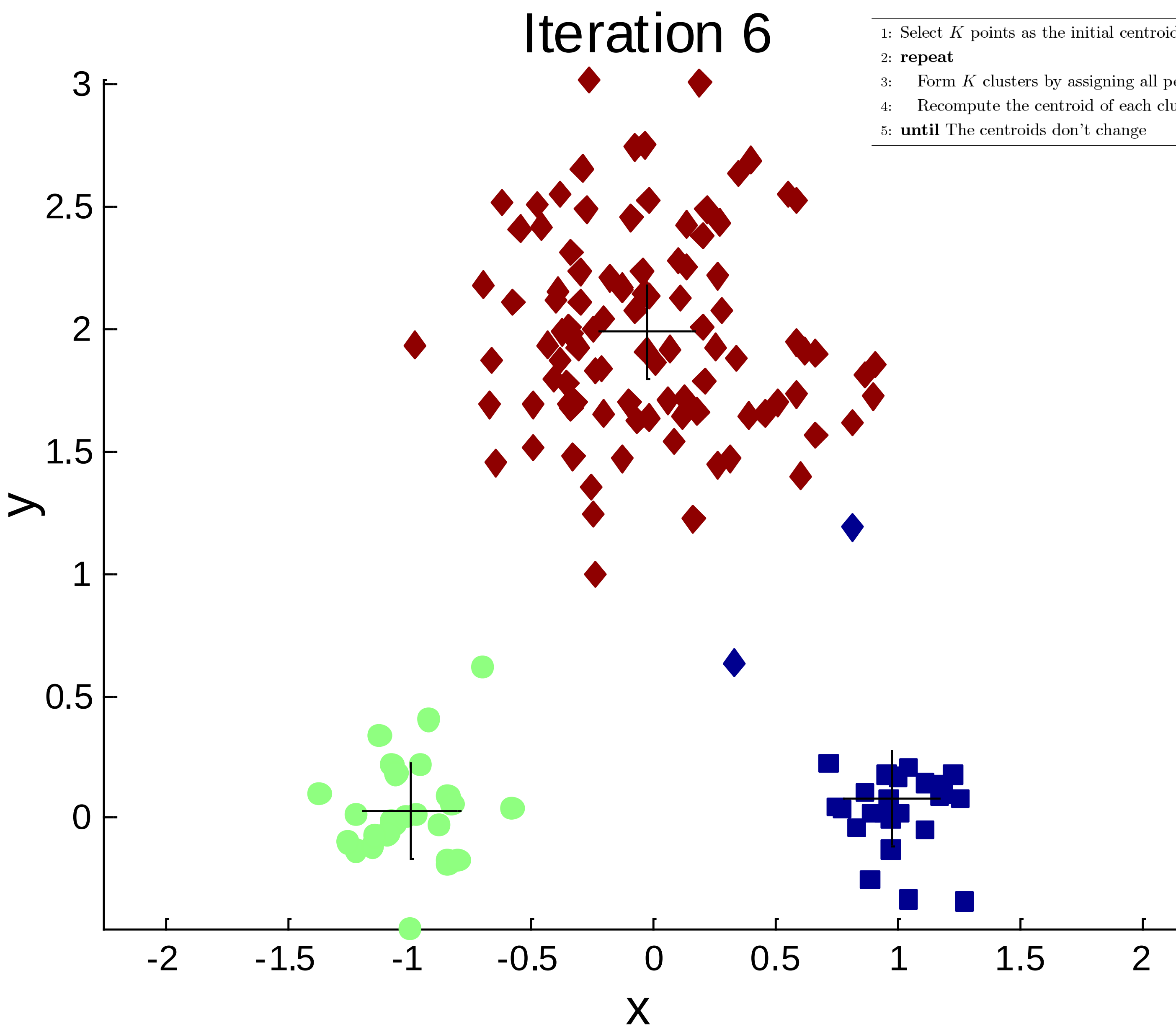
1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change

Example of K-means Clustering

1. Select K points as the initial centroids.
2. **repeat**
3. Form K clusters by assigning all points to the closest centroid.
4. Recompute the centroid of each cluster.
5. **until** The centroids don't change

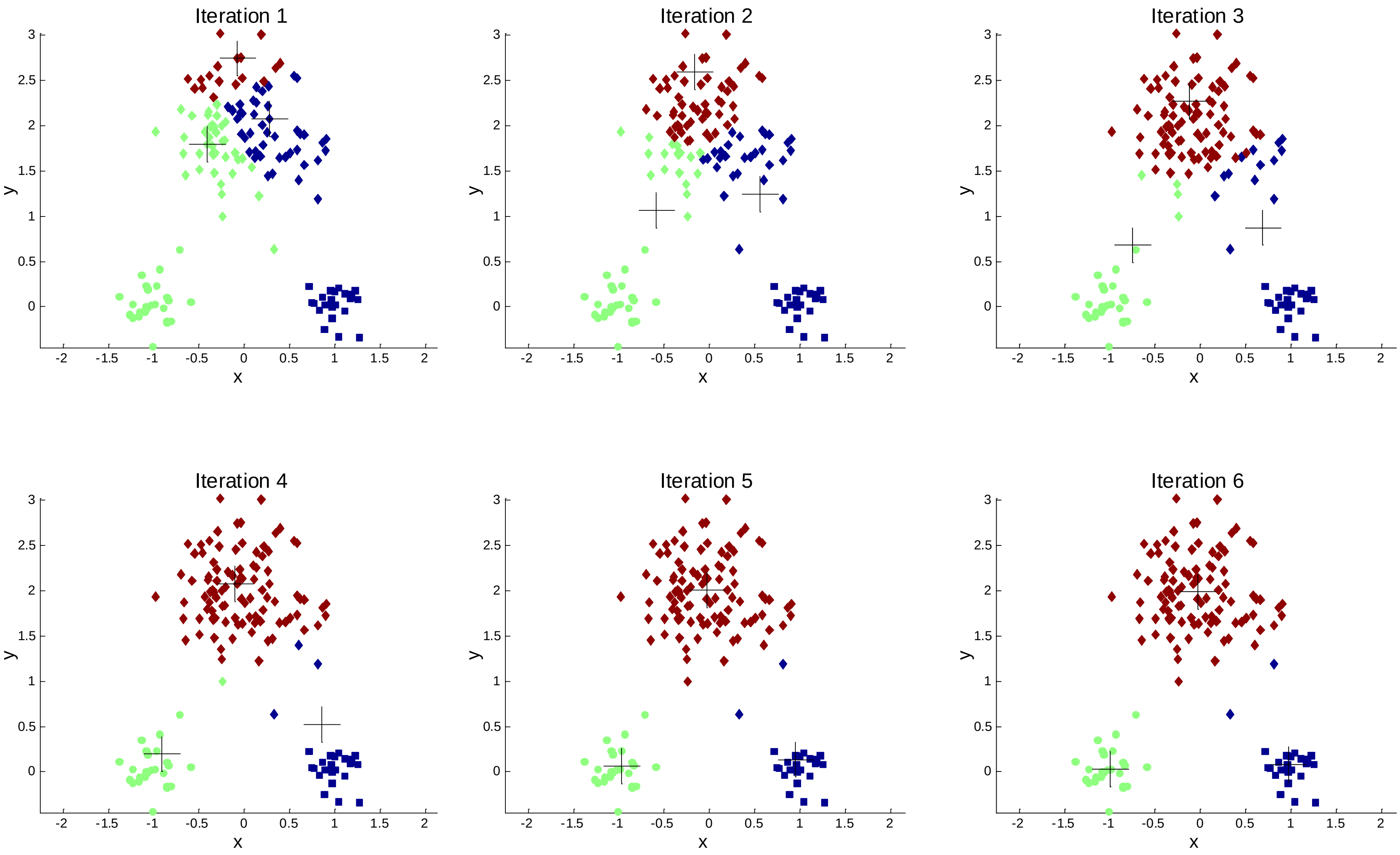


Example of K-means Clustering

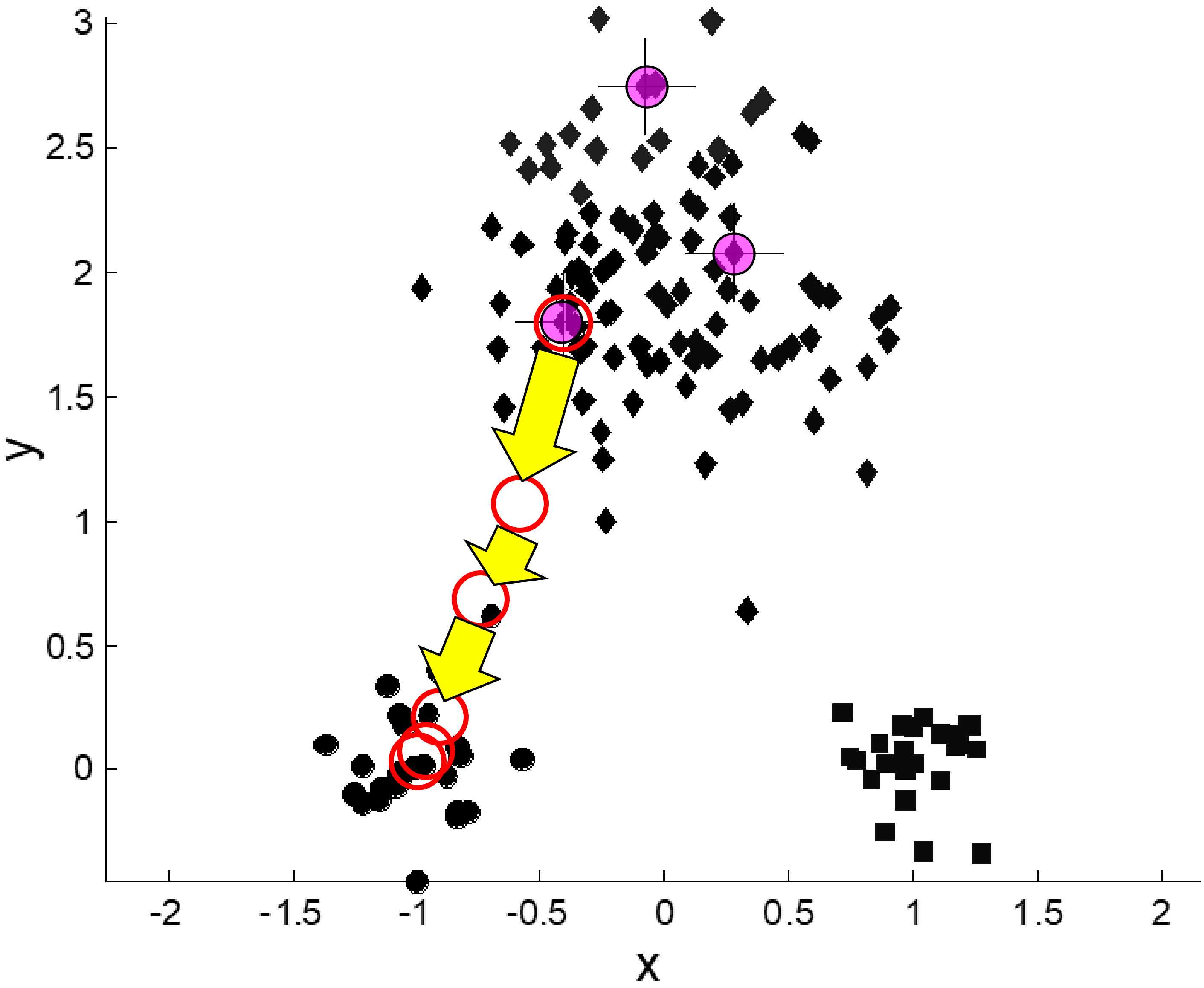



```
Select K points as the initial centroids.
repeat
    Form K clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change
```

Example of K-means Clustering



Example of K-means Clustering



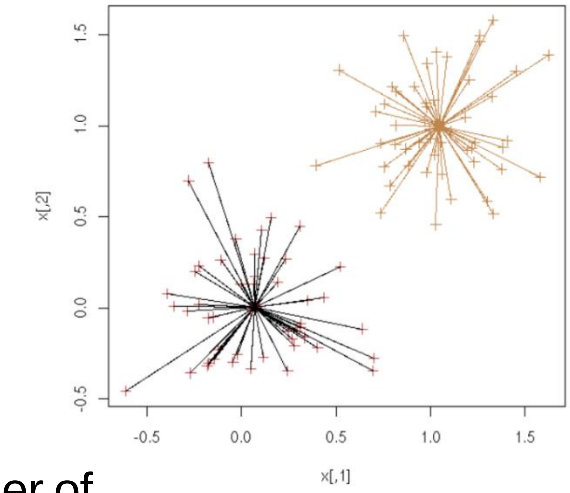
K-means Clustering – Details

- Initial centroids are often chosen randomly.
  - Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster.
- 'Closeness' is measured by Euclidean distance, cosine similarity, correlation, etc.
- K-means will converge for common similarity measures mentioned above.
- 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

Evaluating K-means Clusters

- Most common measure is Sum of Squared Error (SSE)
  - For each point, the error is the distance to the nearest cluster
  - To get SSE, we square these errors and sum them.

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

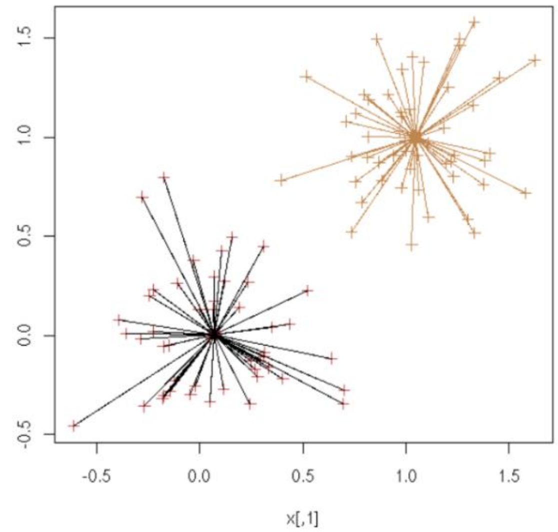


- x is a data point in cluster C_i and m_i is the representative point for cluster C_i
- Given two sets of clusters, we prefer the one with the smallest error
- One easy way to reduce SSE is to increase K, the number of clusters
- A good clustering with smaller K can have a lower SSE than a poor clustering with higher K

K-means as a greedy optimization alg.

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

```
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
```



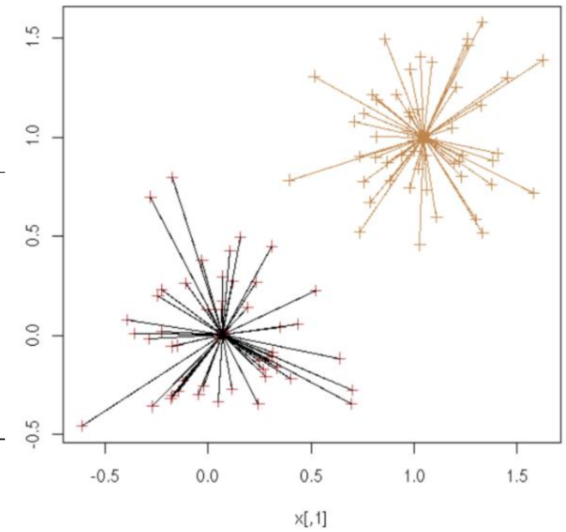
- **Q:** Given K centroids, which cluster assignment minimizes SSE ?
  - **A:** Each object contributes to the SSE; just minimize each contribution, i.e. assign the centroid with smallest dist(m_i,x)

- **Q:** Given a set of clusters, which centroids minimize SSE ?
- **A:** Assume Euclidean distance; minimize by cluster; set partial derivatives to zero (for all "j": d SSE / d m[j] = 0) and solve. Result: m_i = mean point of the cluster.

K-means as a greedy optimization alg.

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

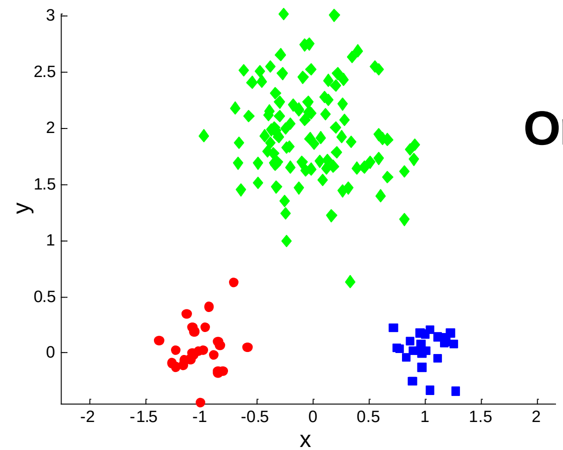
1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change



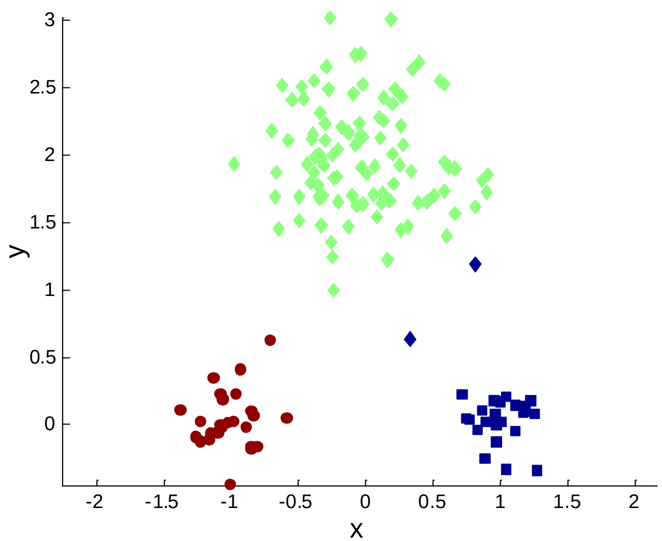
- **Q:** Does k-means terminate ?
  - **A:** Yes. At each step SSE cannot increase, and the number of clustering configurations is finite, thus will stop, sooner or later (Tricky examples exist. Better replace 5. with "SSE does not decrease")

- **Q:** Does k-means find the optimum ?
- **A:** It finds a local optimum, which might be not the global one (see next slides for examples)

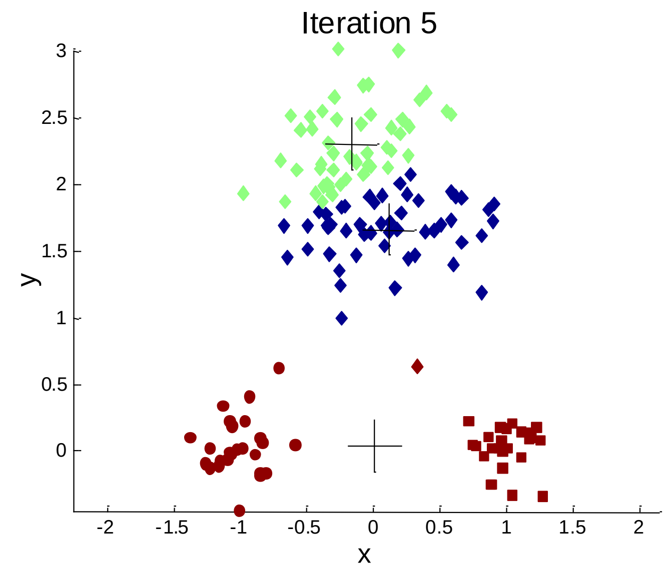
Two different K-means Clusterings



Original Points

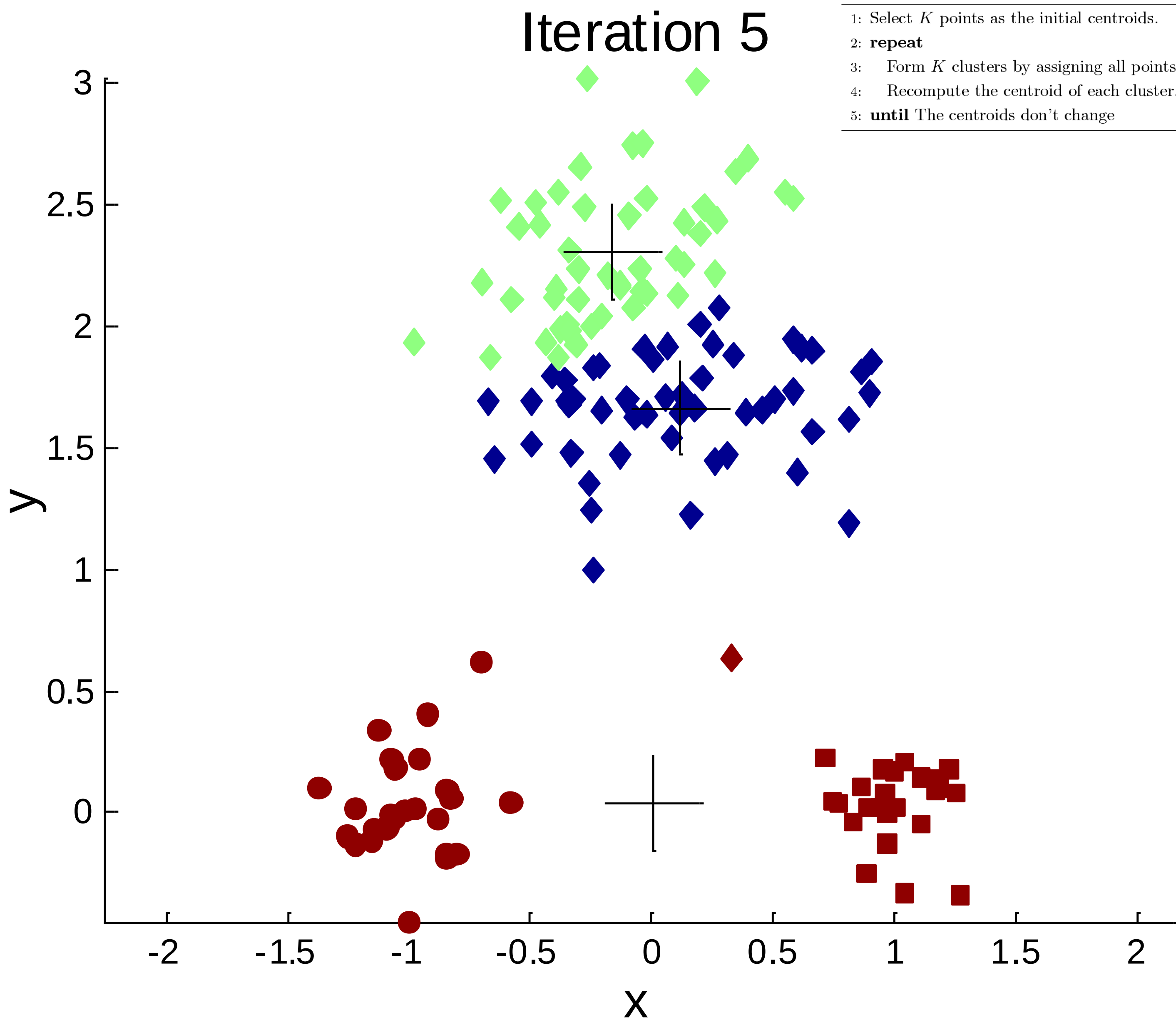


Optimal Clustering



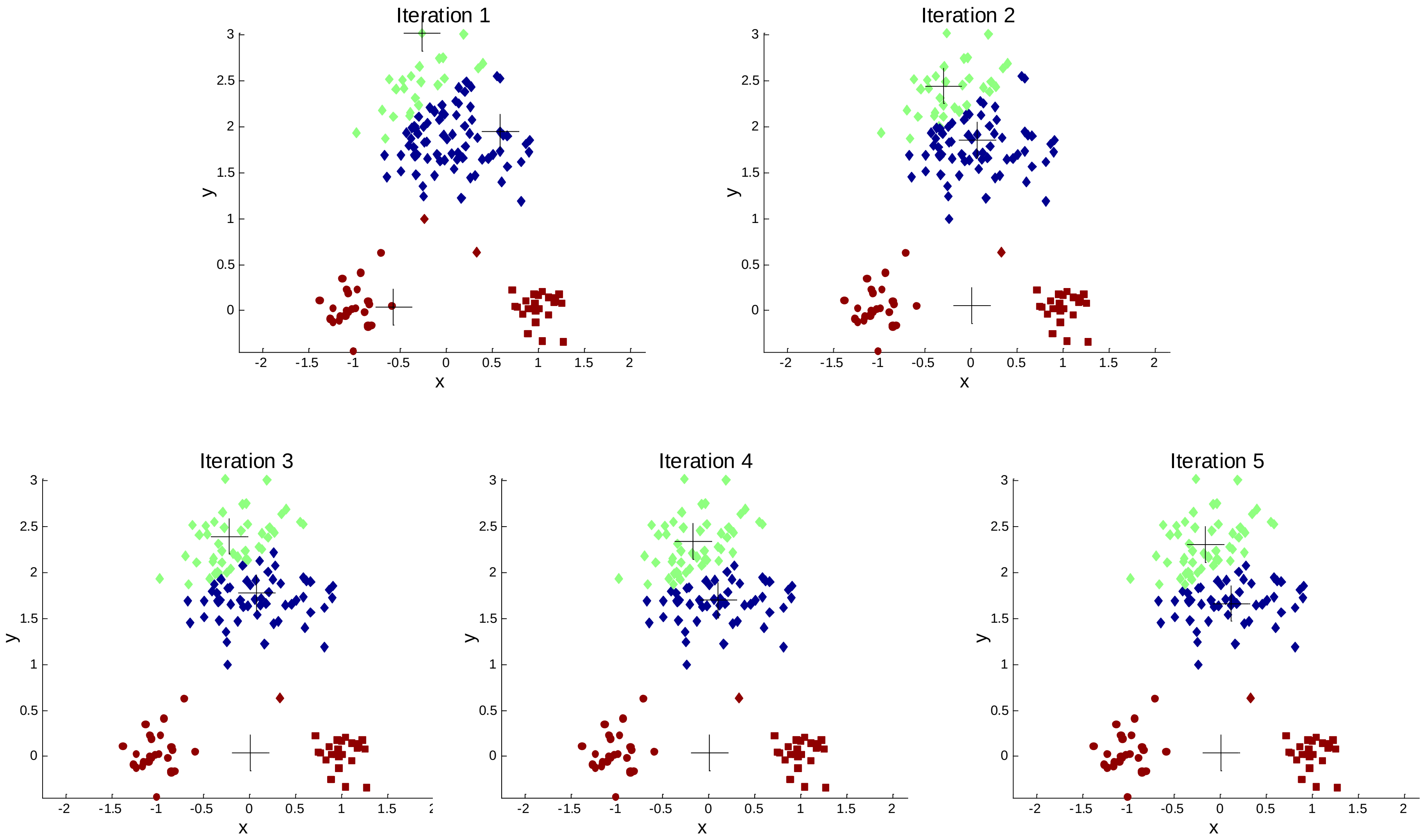
Sub-optimal Clustering

Importance of Choosing Initial Centroids …




```
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
```

Importance of Choosing Initial Centroids …

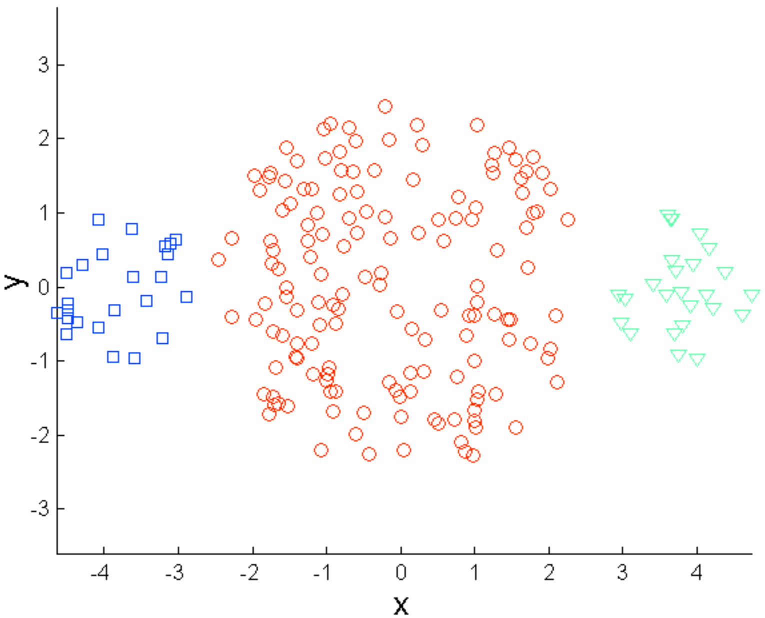


Limitations of K-means

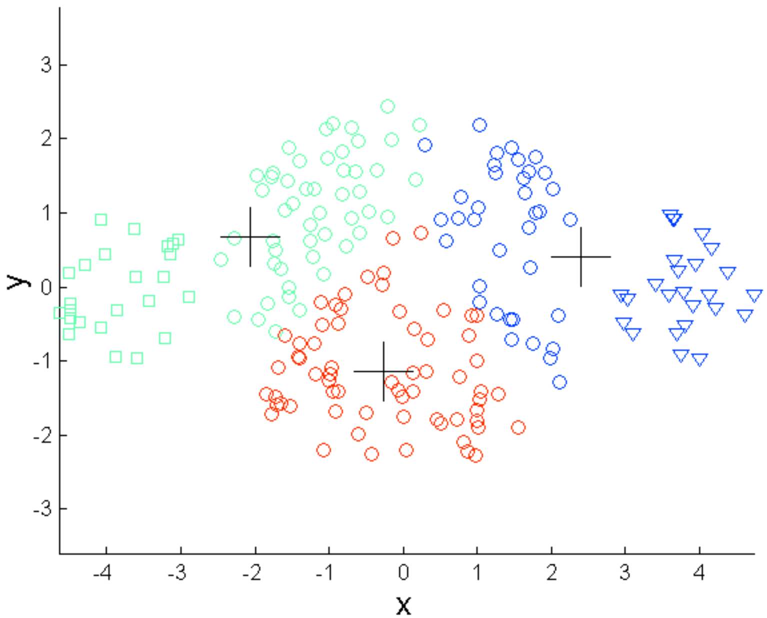
- K-means has problems when clusters are of differing
  - Sizes
  - Densities
  - Non-globular shapes

- K-means has problems when the data contains outliers.

Limitations of K-means: Differing Sizes

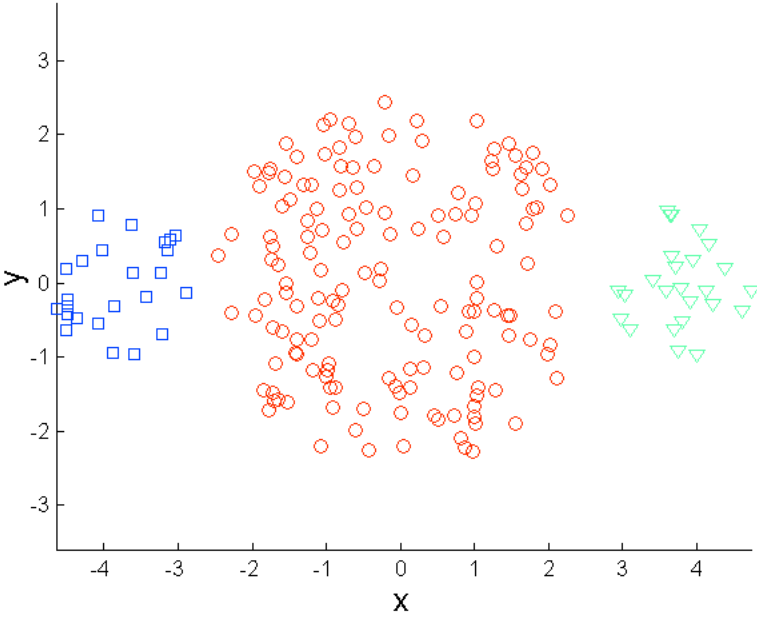


Original Points

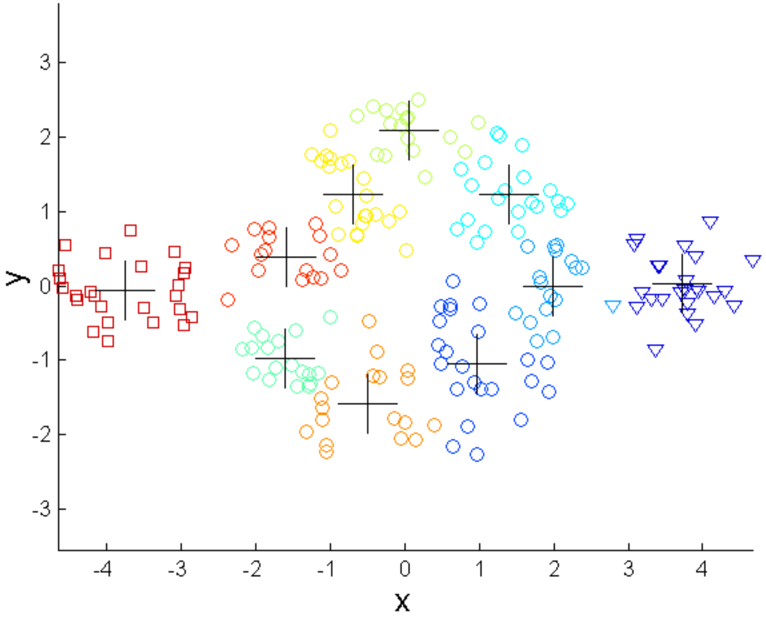


K-means (3 Clusters)

Overcoming K-means Limitations



Original Points

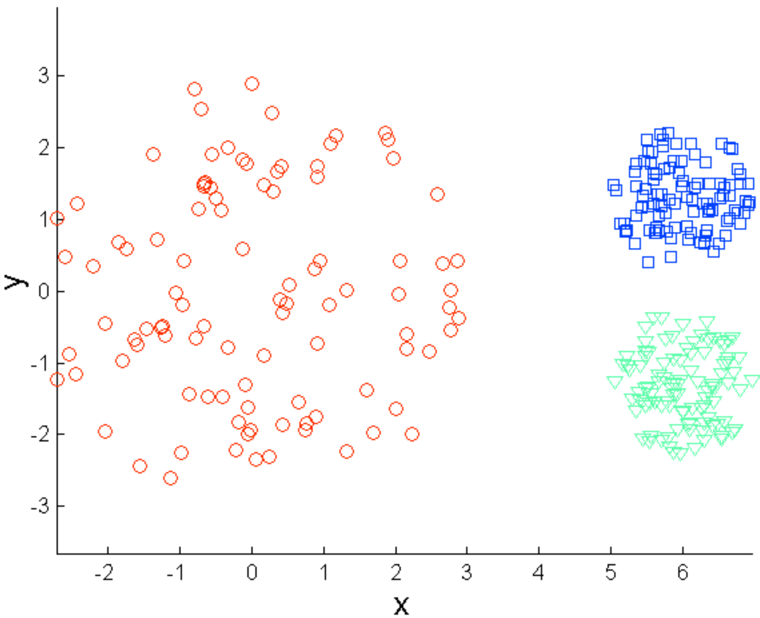


K-means Clusters

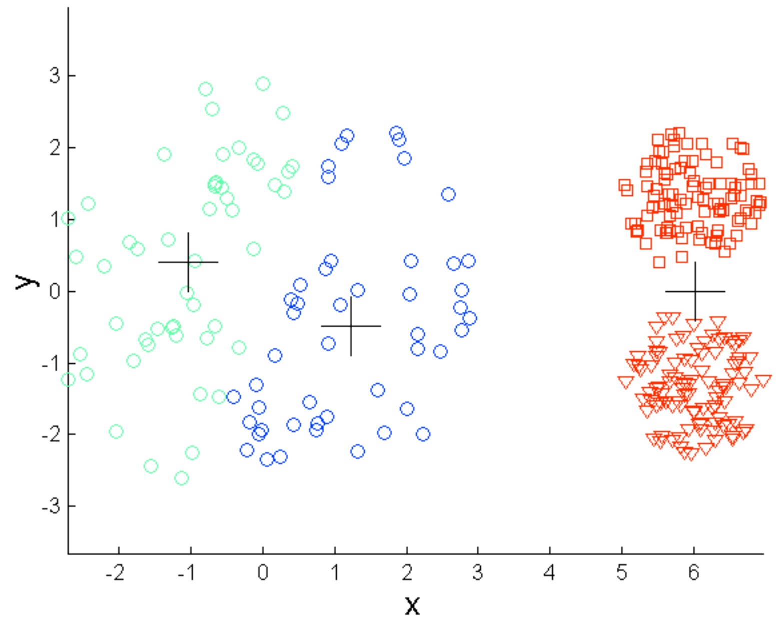
One solution is to use many clusters.

Find parts of clusters, but need to put together.

Limitations of K-means: Differing Density

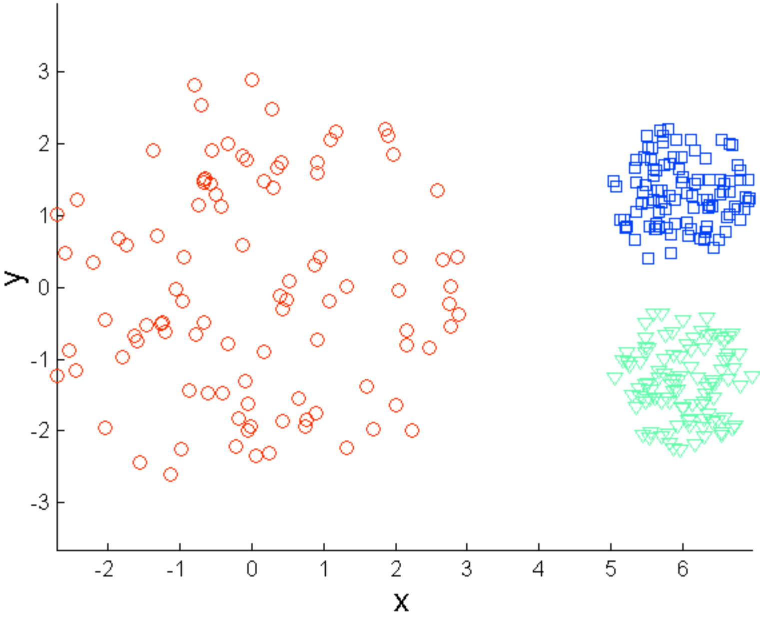


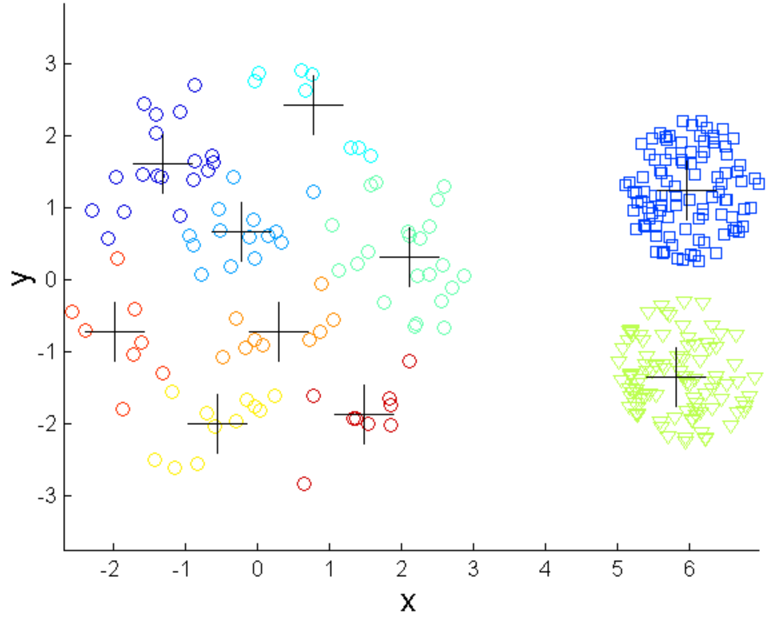
Original Points



K-means (3 Clusters)

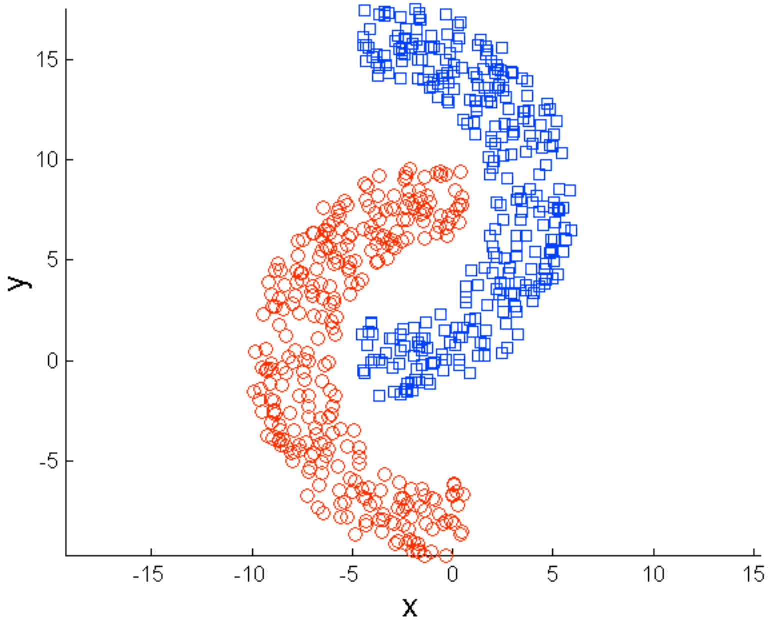
Overcoming K-means Limitations



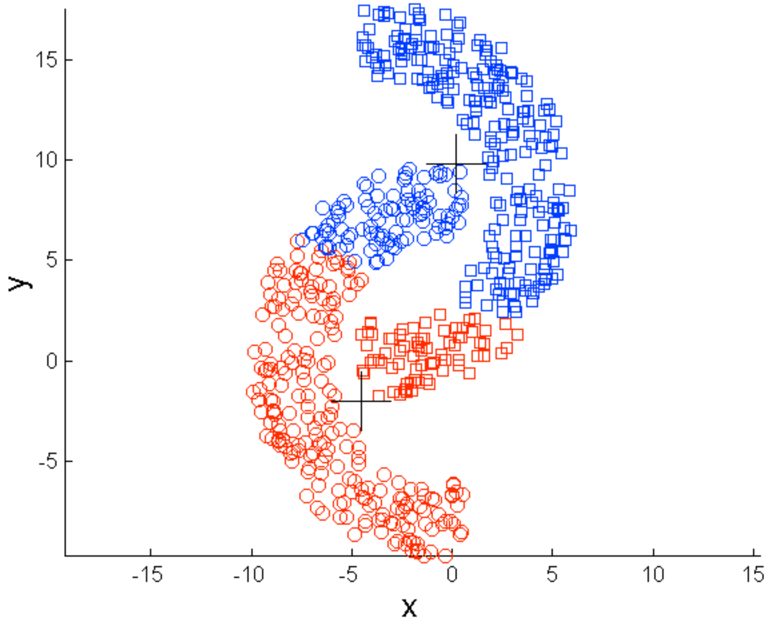


Original Points **K-means Clusters**

Limitations of K-means: Non-globular Shapes

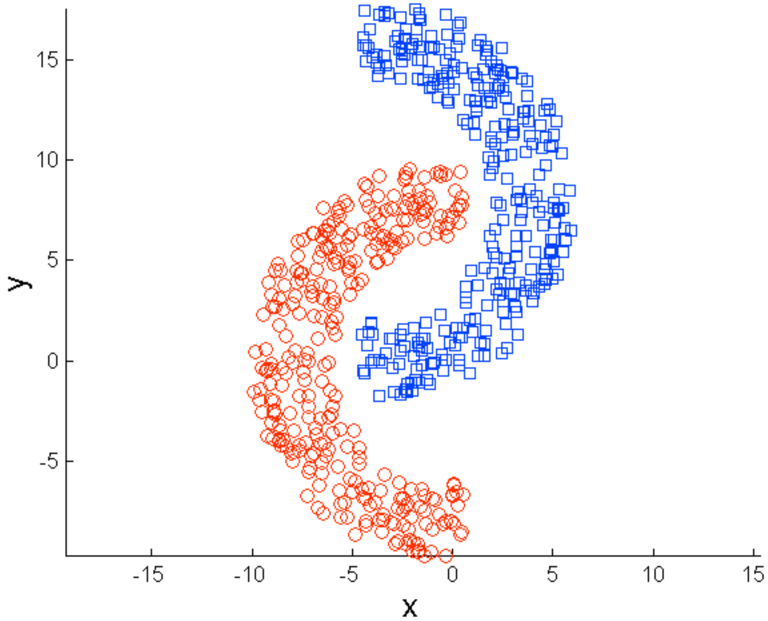


Original Points

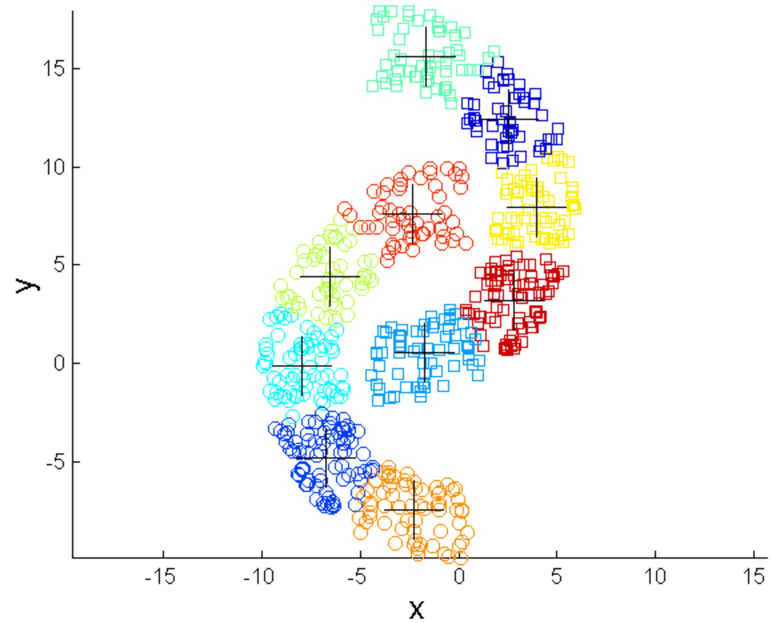


K-means (2 Clusters)

Overcoming K-means Limitations



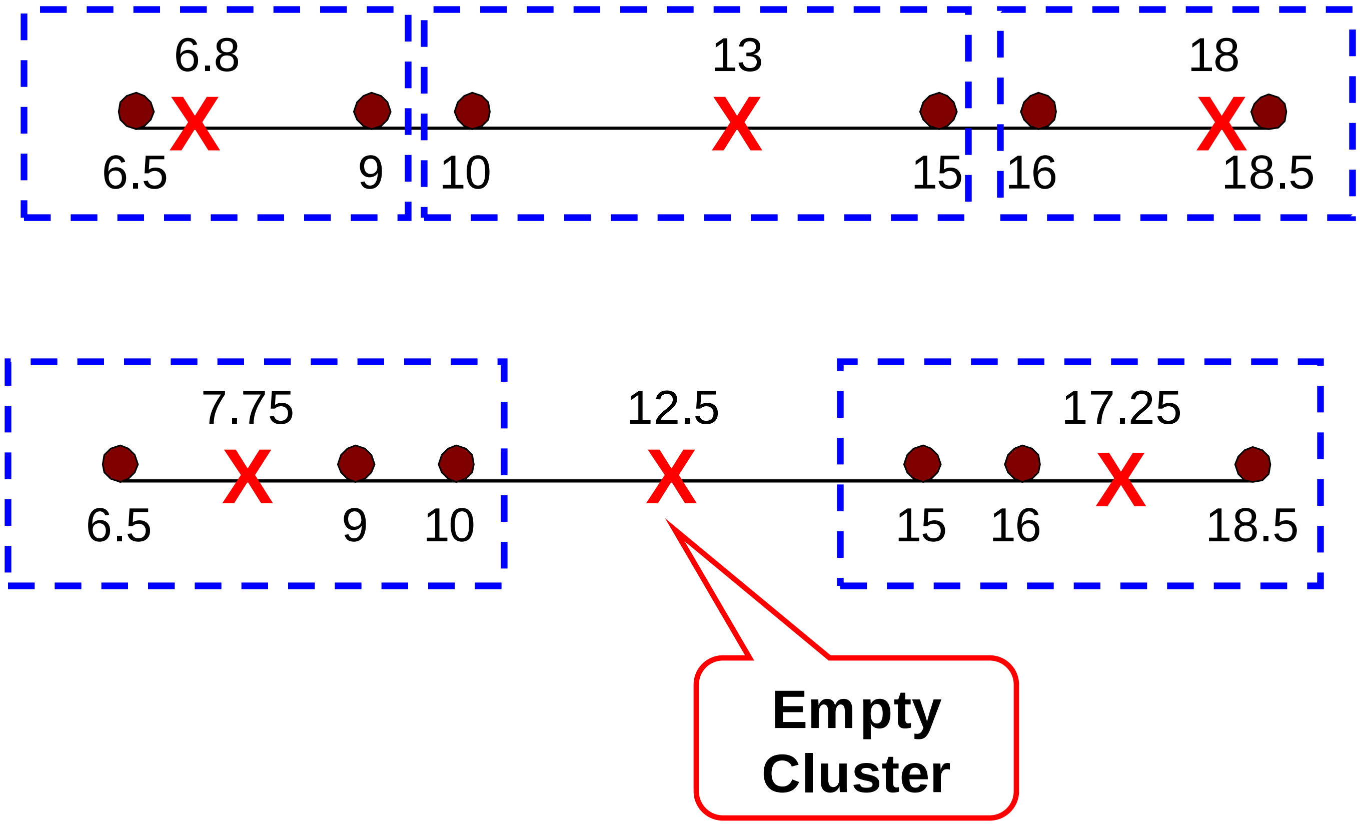
Original Points



K-means Clusters

Empty Clusters

- K-means can yield empty clusters



Handling Empty Clusters

- Basic K-means algorithm can yield empty clusters

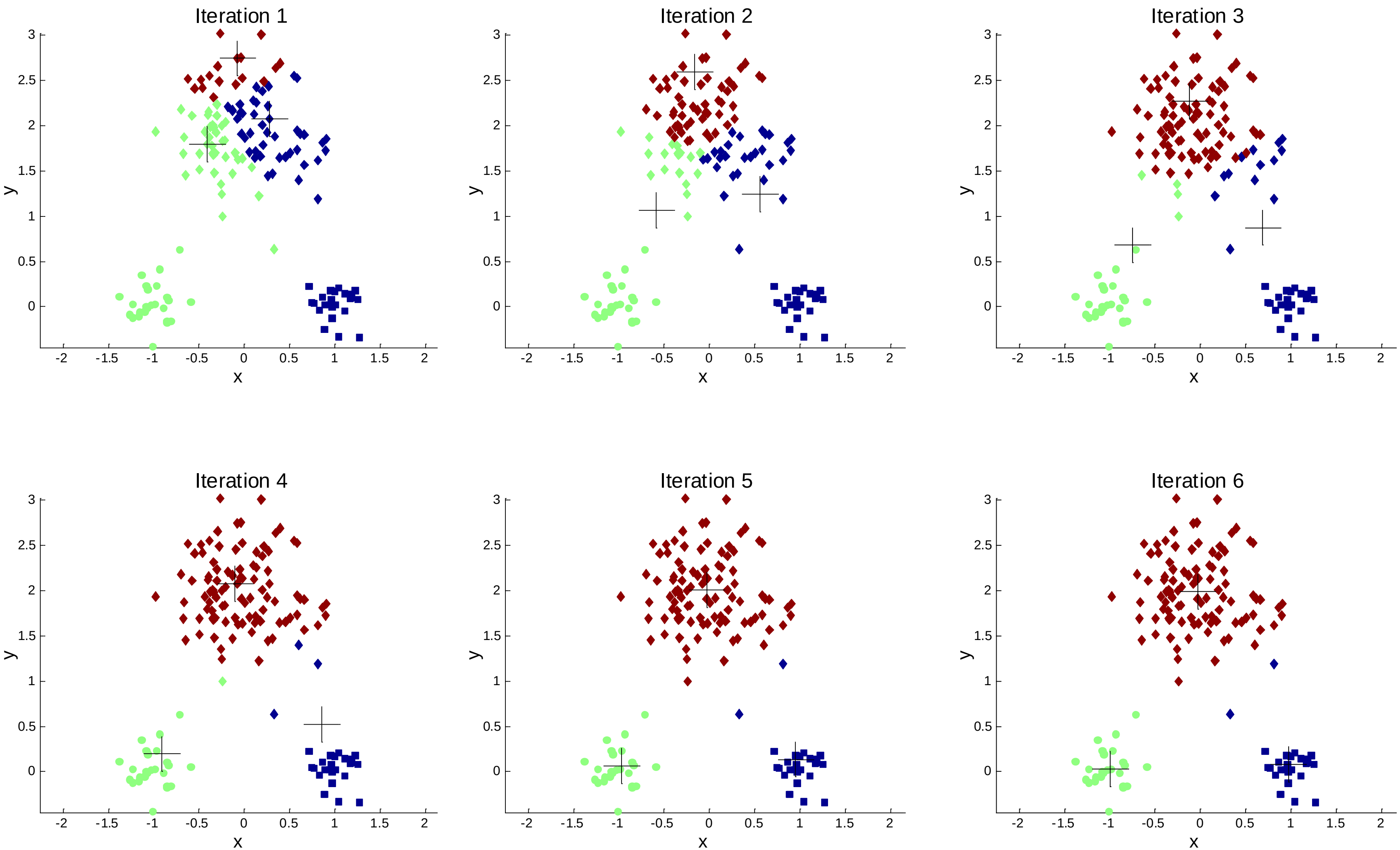
- Several strategies
  - Choose a point and assign it to the cluster
    - Choose the point that contributes most to SSE
    - Choose a point from the cluster with the highest SSE

- If there are several empty clusters, the above can be repeated several times.

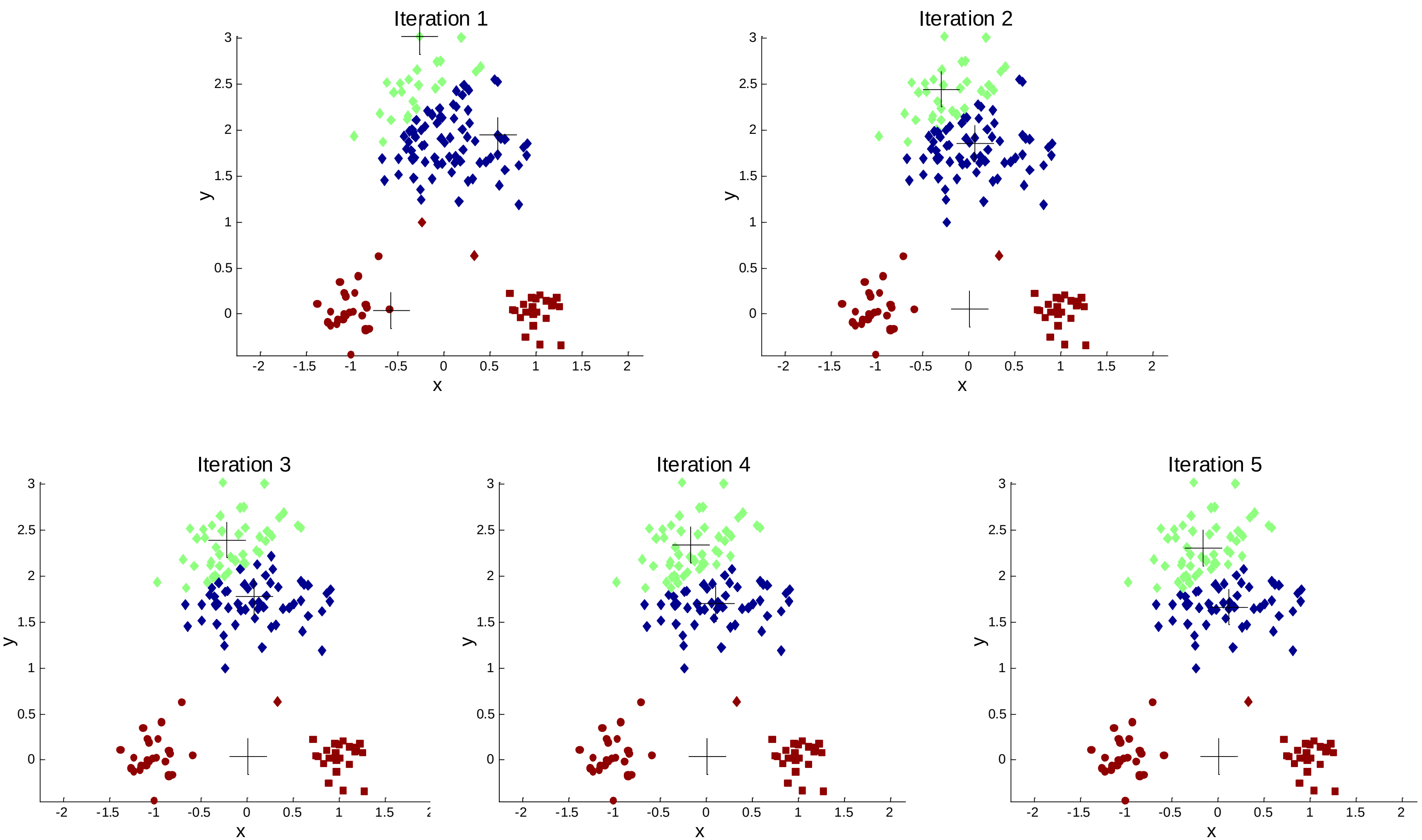
Pre-processing and Post-processing

- Pre-processing
  - Normalize the data
  - Eliminate outliers
- Post-processing
  - Eliminate small clusters that may represent outliers
  - Split 'loose' clusters, i.e., clusters with relatively high SSE
  - Merge clusters that are 'close' and that have relatively low SSE
  - Can use these steps during the clustering process
    - ISODATA

Importance of Choosing Initial Centroids



Importance of Choosing Initial Centroids …



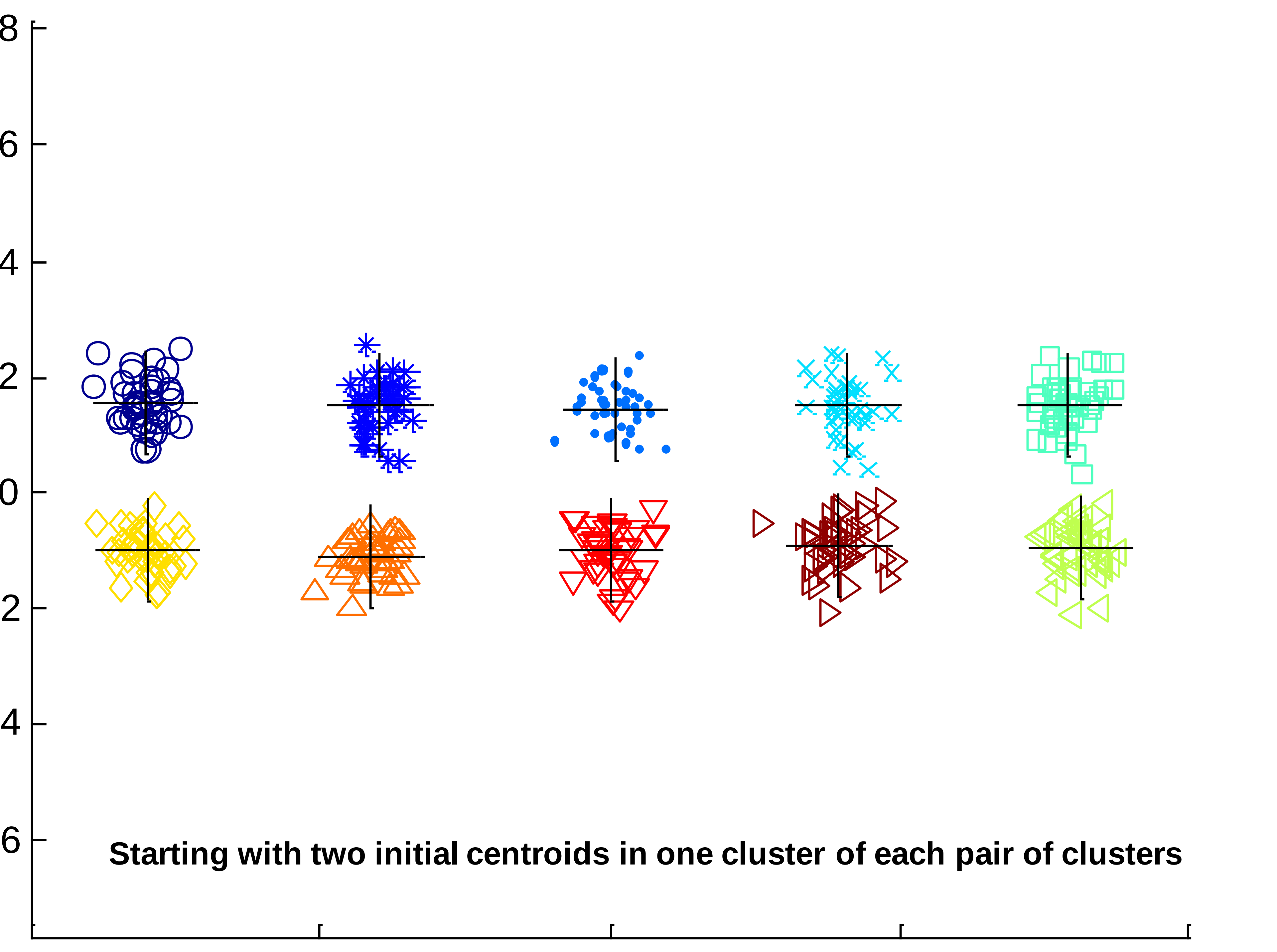
Ideal solution: one initial centroid per cluster

- If there are K 'real' clusters then the chance of selecting one centroid from each cluster is small.
  - Chance is relatively small when K is large
  - If clusters are the same size, n, then

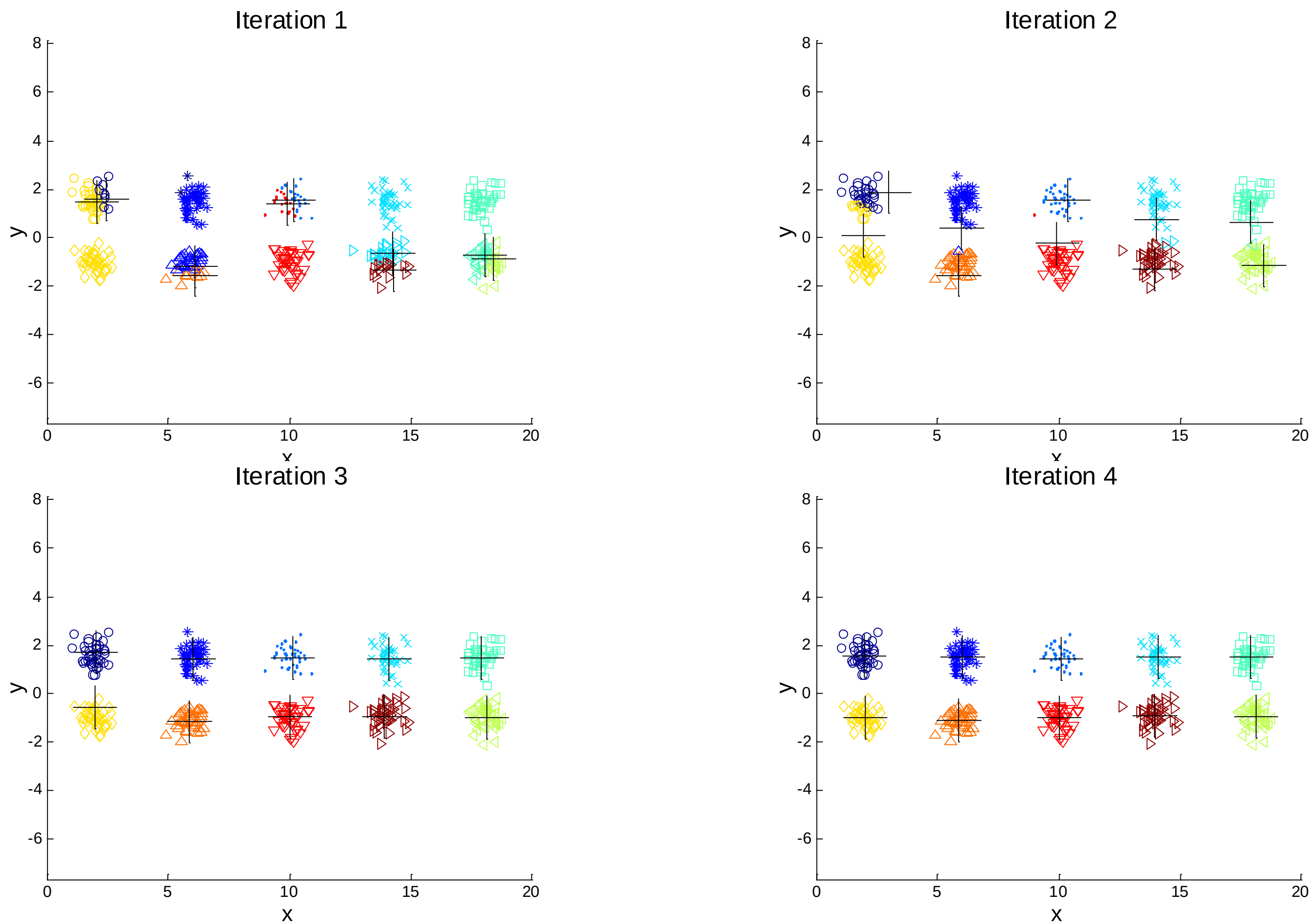
$$P = \frac{\text{number of ways to select one centroid from each cluster}}{\text{number of ways to select } K \text{ centroids}} = \frac{K!n^K}{(Kn)^K} = \frac{K!}{K^K}$$

(assumes the same centroid can be selected multiple times...)

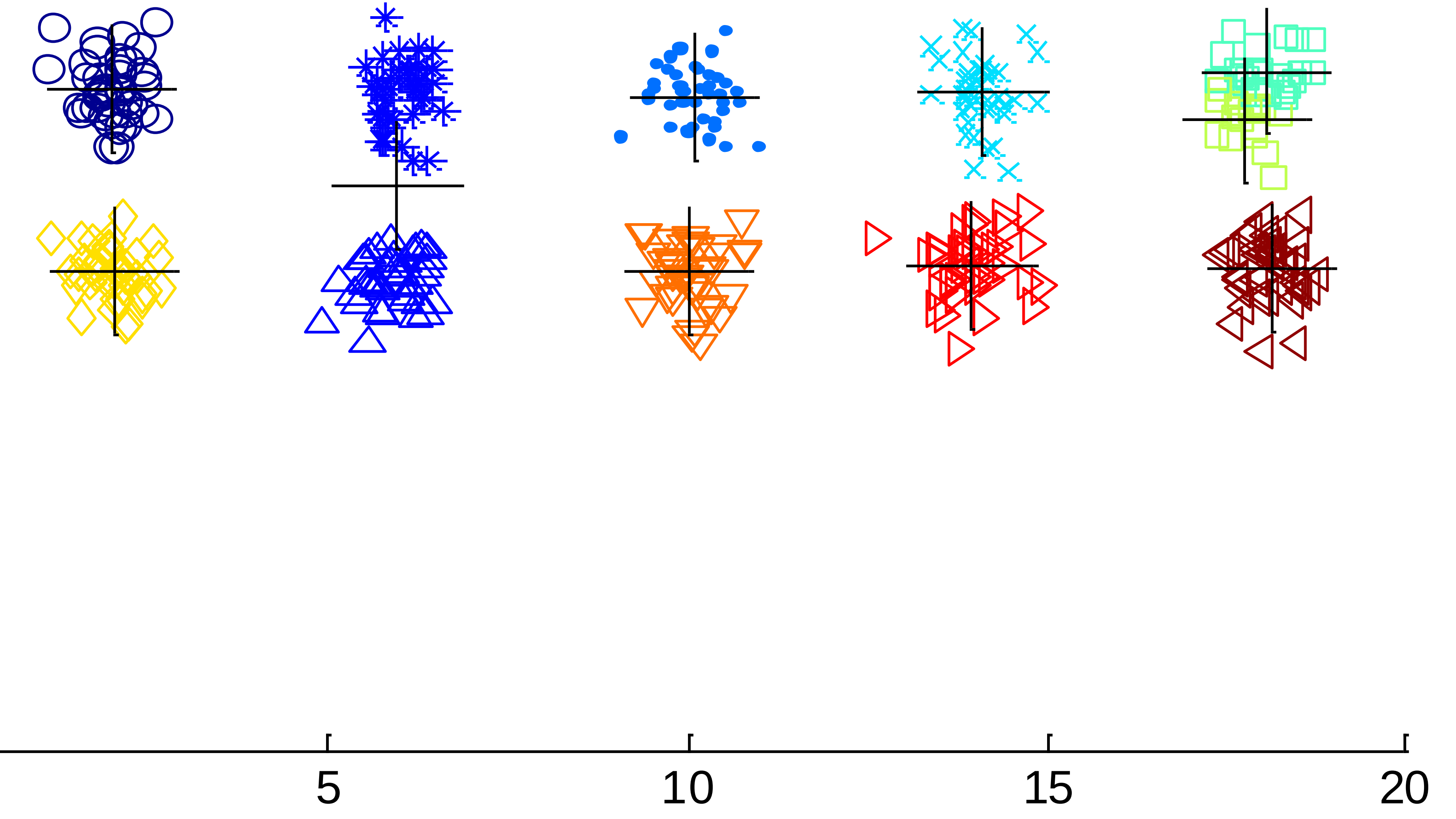
  - For example, if K = 10, then
    - probability = $10!/10^{10} = 0.00036$
  - Sometimes the initial centroids will readjust themselves in 'right' way, and sometimes they don't
  - Consider an example of five pairs of clusters


8
6
4
2
0
2
4
6
Starting with two initial centroids in one cluster of each pair of clusters

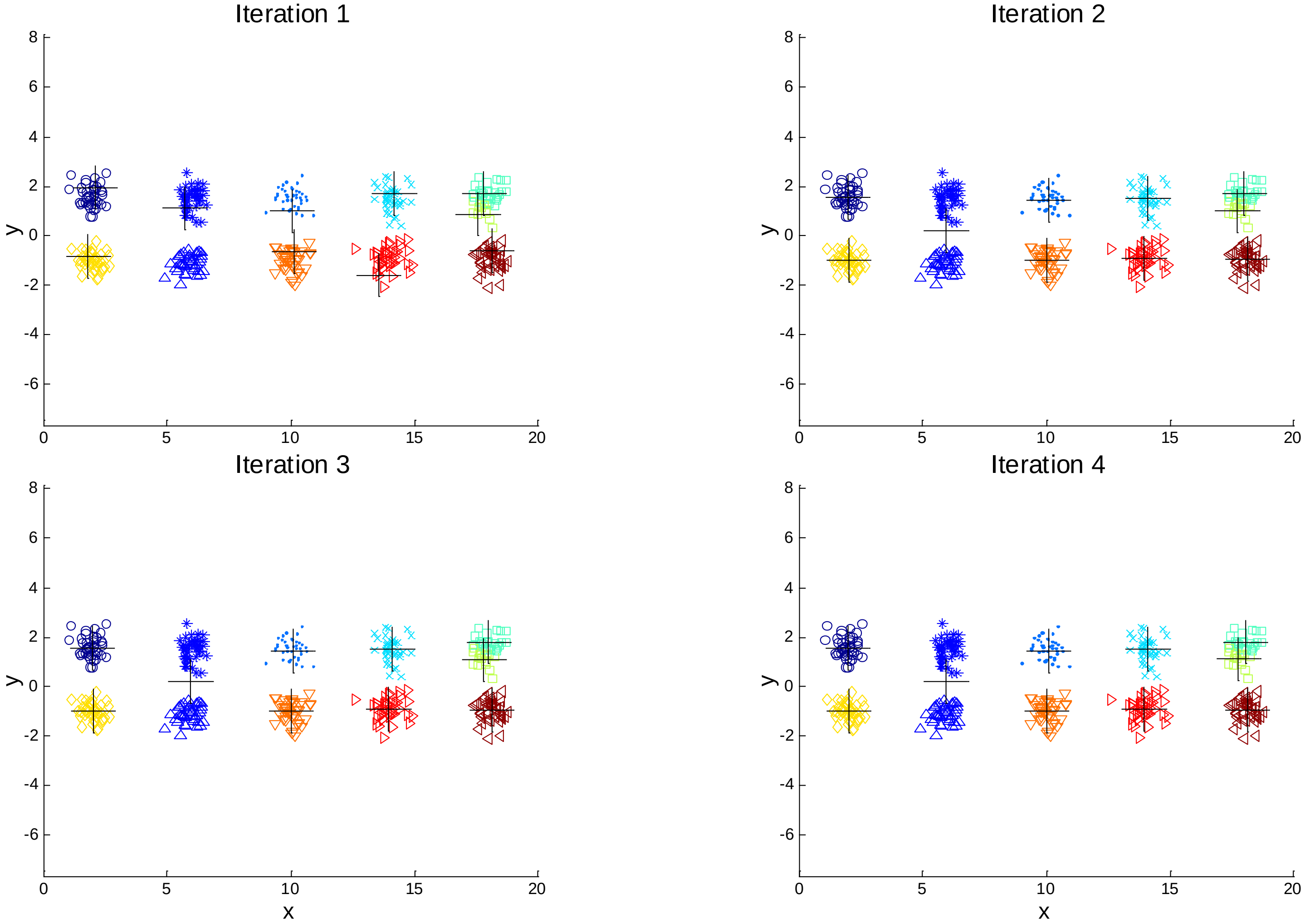
10 Clusters Example



Starting with two initial centroids in one cluster of each pair of clusters


5
10
15
20

10 Clusters Example



Starting with some pairs of clusters having three initial centroids, while other have only one.

Solutions to Initial Centroids Problem

- Multiple runs
  - Helps, but probability is not on your side
- Extract a small sample of data and use hierarchical clustering to determine initial centroids
- Select more than k initial centroids and then select among these initial centroids
  - Select most widely separated
- Post-processing
- Generate a larger number of clusters and then group them -- e.g. by a hierarchical clustering
- Bisecting K-means
  - Not as susceptible to initialization issues

Updating Centers Incrementally

- In the basic K-means algorithm, centroids are updated after all points are assigned to a centroid

- An alternative is to update the centroids after each assignment (incremental approach)
  – Each assignment updates zero or two centroids
  – More expensive
  – Introduces an order dependency
  – Never get an empty cluster
  – Can use "weights" to change the impact

Algorithm 3 Bisecting K-means Algorithm.

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
until Until the list of clusters contains K clusters

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

Bisecting K-means Example