

Clustering I

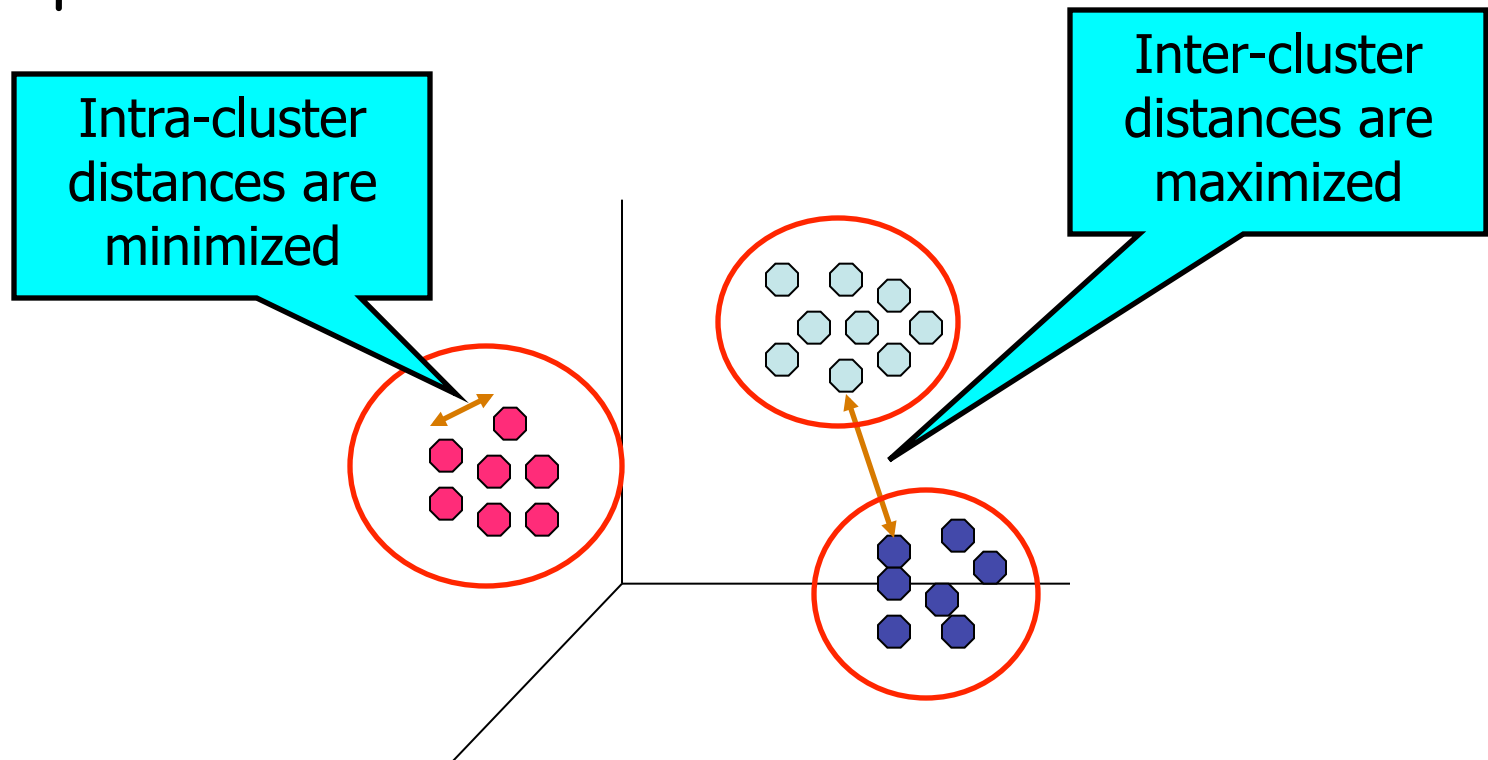
Wei Pang

The Task

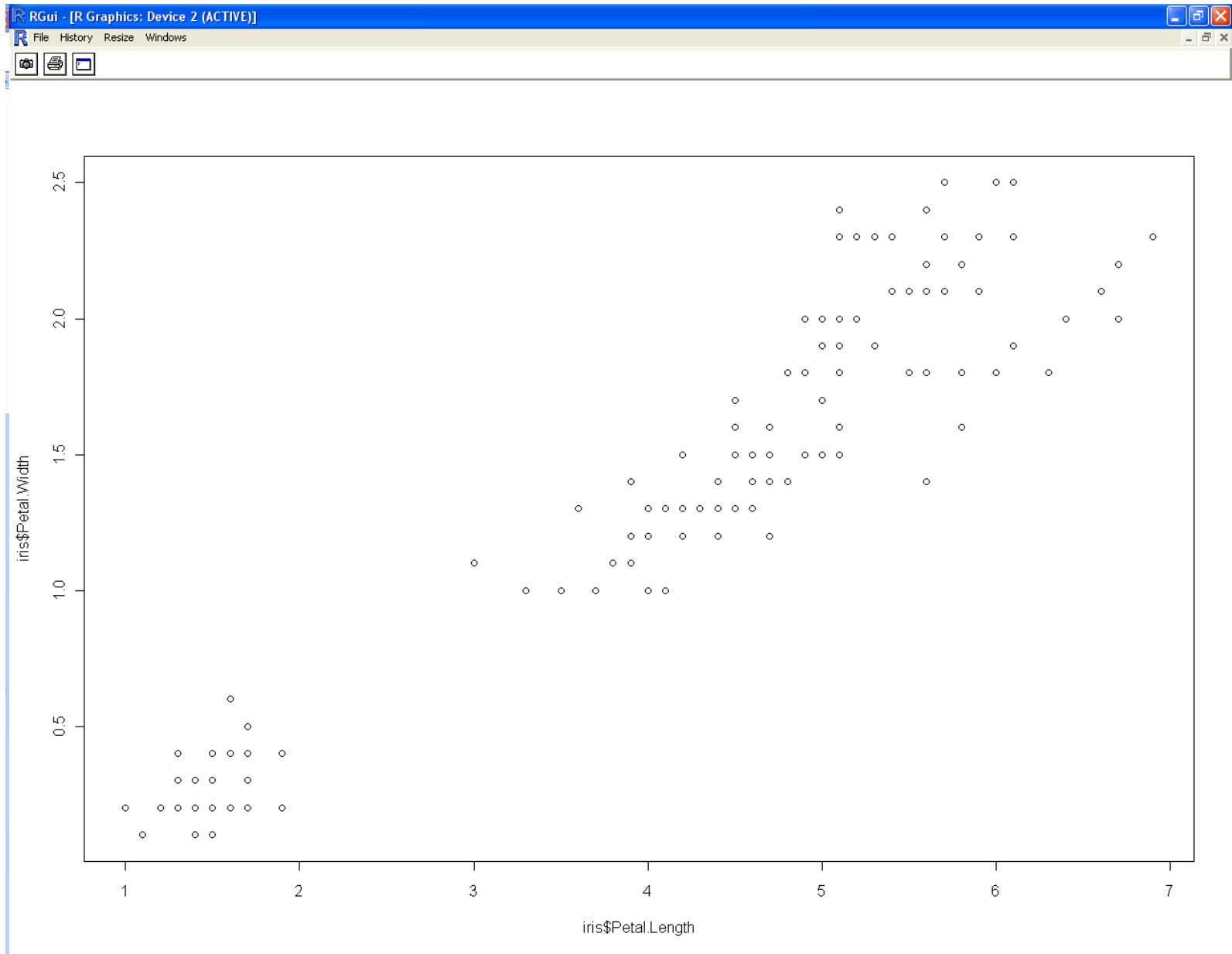
- Input: Collection of instances
 - No special class label attribute!
- Output: Clusters (Groups) of instances where members of a cluster are more 'similar' to each other than they are to members of another cluster
 - Similarity between instances need to be defined
- Because there are no predefined classes to predict, the task is to find useful groups of instances

What is Cluster Analysis?

- 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

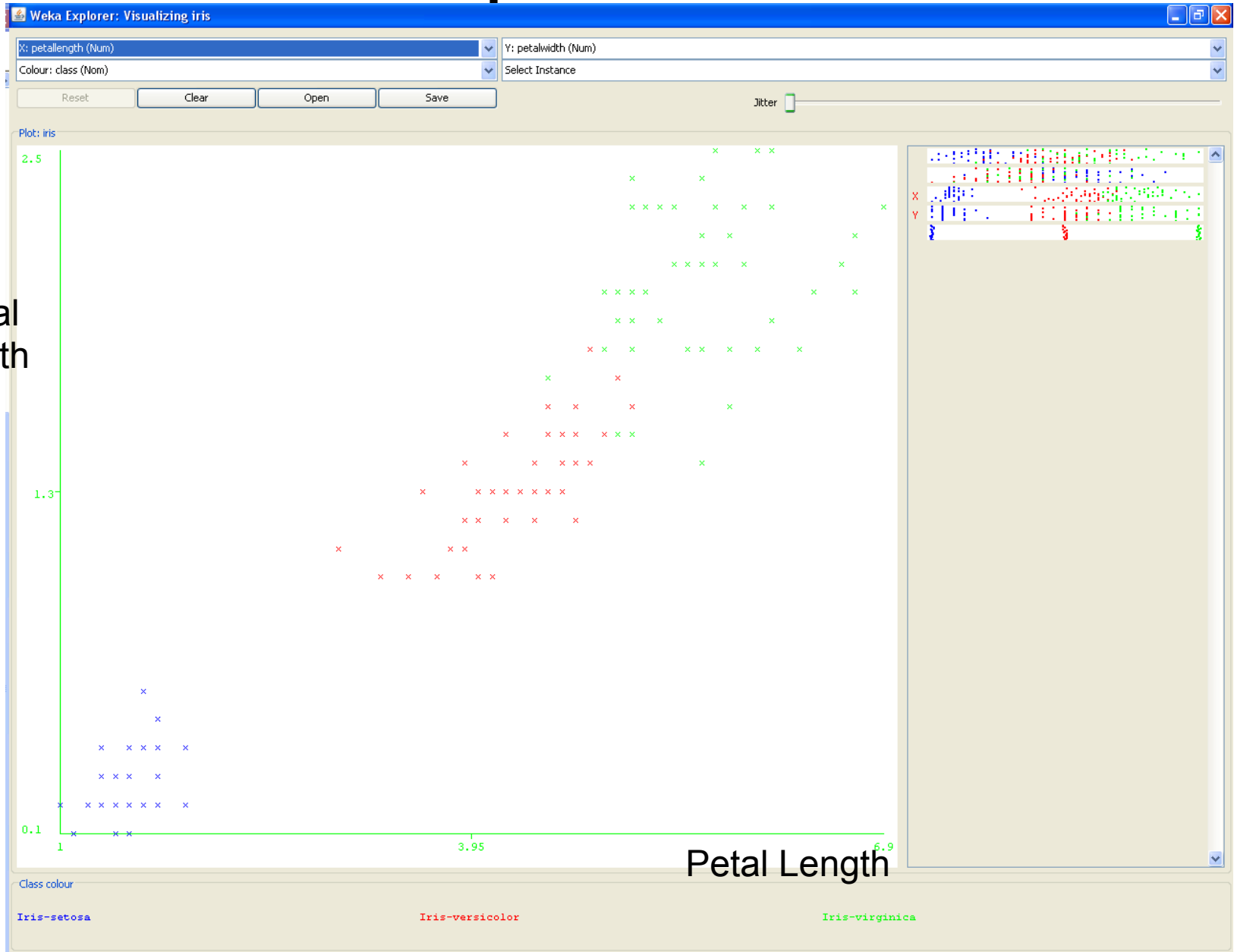


Example - Iris Data



Example Clusters?

Petal
Width

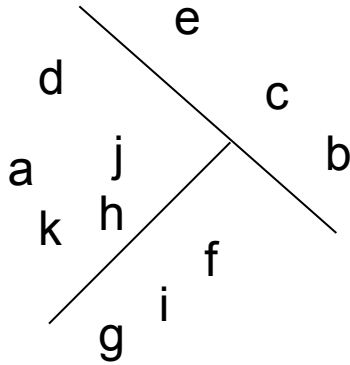


What is not Cluster Analysis?

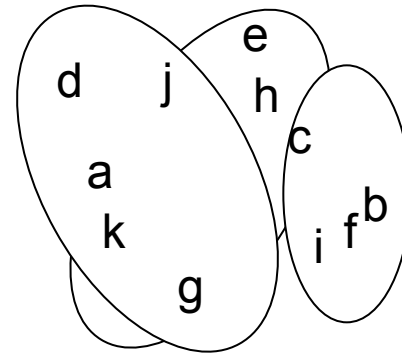
- Supervised classification
 - Have class label information
- Simple segmentation
 - Dividing students into different registration groups alphabetically, by last name
- Results of a query
 - Groupings are a result of an external specification
- Graph partitioning
 - Some mutual relevance and synergy, but areas are not identical

Cluster Representations

- Several representations are possible



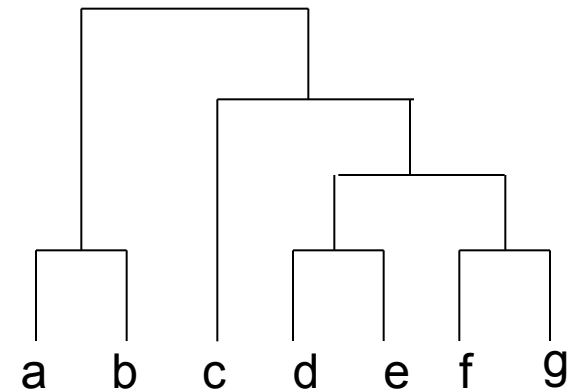
Partitioning 2D space showing instances



Venn Diagram

	1	2	3
a	0.4	0.1	0.5
b	0.1	0.8	0.1
c	0.3	0.3	0.4
d	0.1	0.1	0.8

Table of Cluster Probabilities

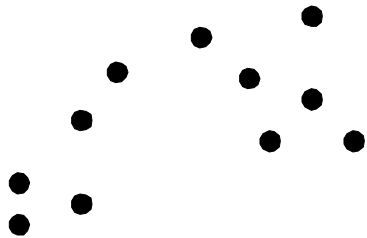


Dendrogram

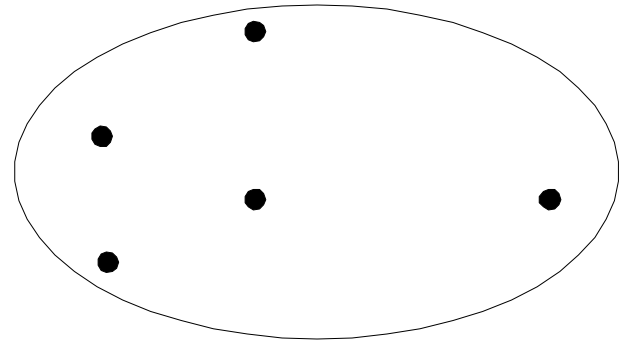
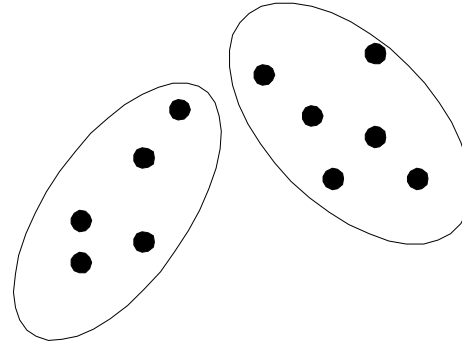
Types of Clusterings

- A **clustering** is a set of clusters
- Important distinction between **hierarchical** and **partitional** sets of clusters
- Partitional Clustering
 - A division 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

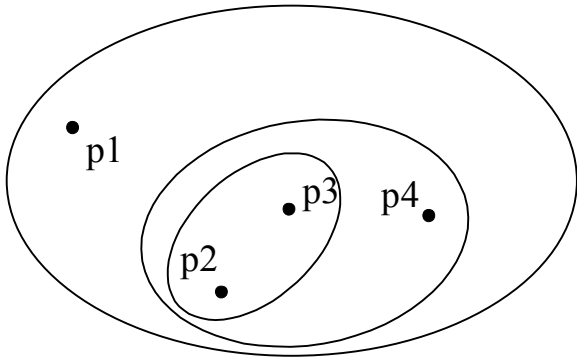


Original Points

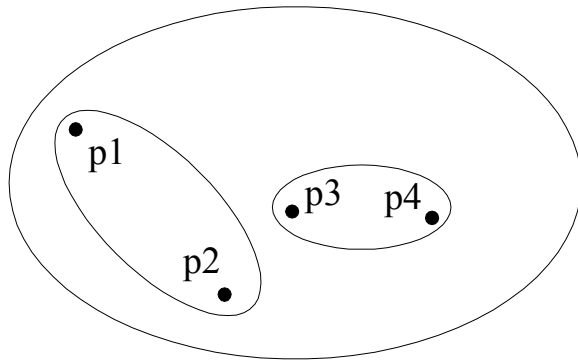


A Partitional Clustering

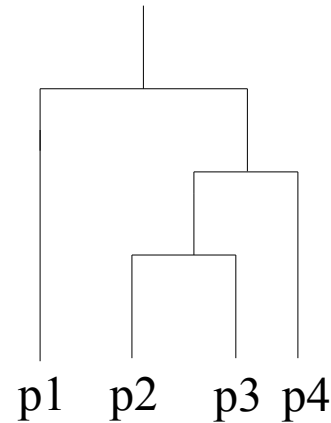
Hierarchical Clustering



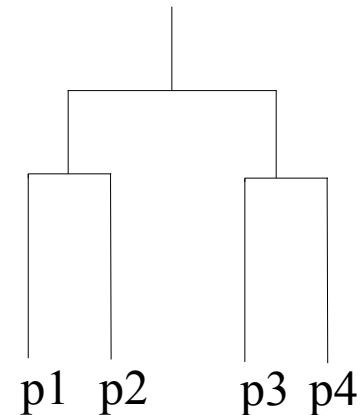
Traditional Hierarchical Clustering



Non-traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Dendrogram

Other Distinctions Between Sets of Clusters

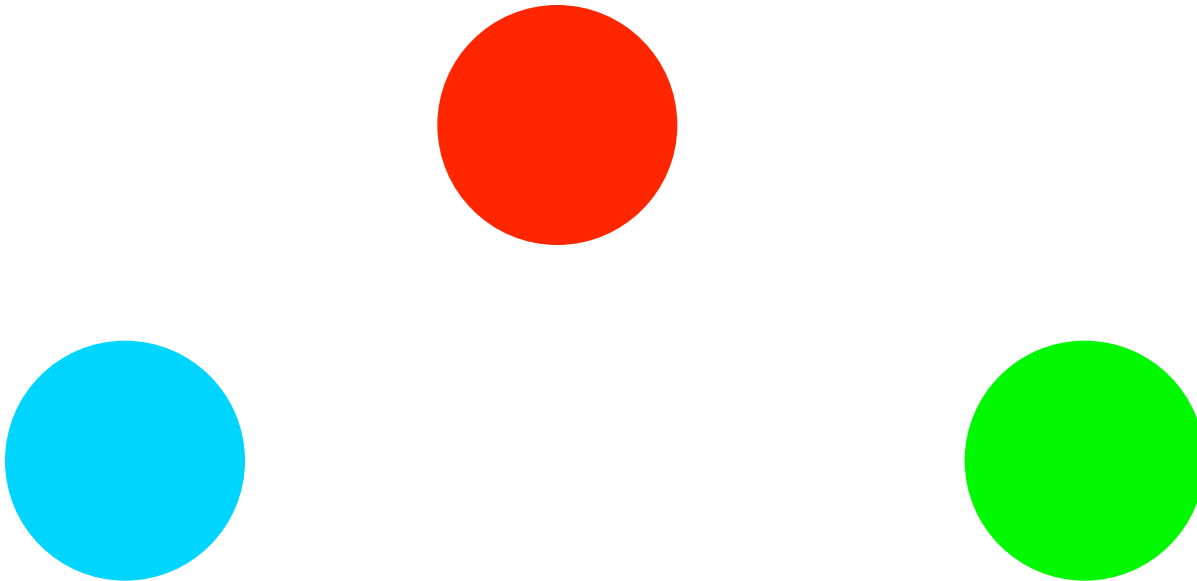
- **Exclusive versus non-exclusive**
 - In non-exclusive clusterings, points may belong to multiple clusters.
 - Can represent multiple classes or 'border' points
- **Fuzzy versus non-fuzzy**
 - In fuzzy clustering, a point belongs to every cluster with some weight between 0 and 1
 - Weights must sum to 1
 - Probabilistic clustering has similar characteristics
- **Partial versus complete**
 - In some cases, we only want to cluster some of the data
- **Heterogeneous versus homogeneous**
 - Cluster of widely different sizes, shapes, and densities

Types of Clusters

- Well-separated clusters
- Center-based clusters
- Contiguous clusters
- Density-based clusters
- Property or Conceptual
- 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.

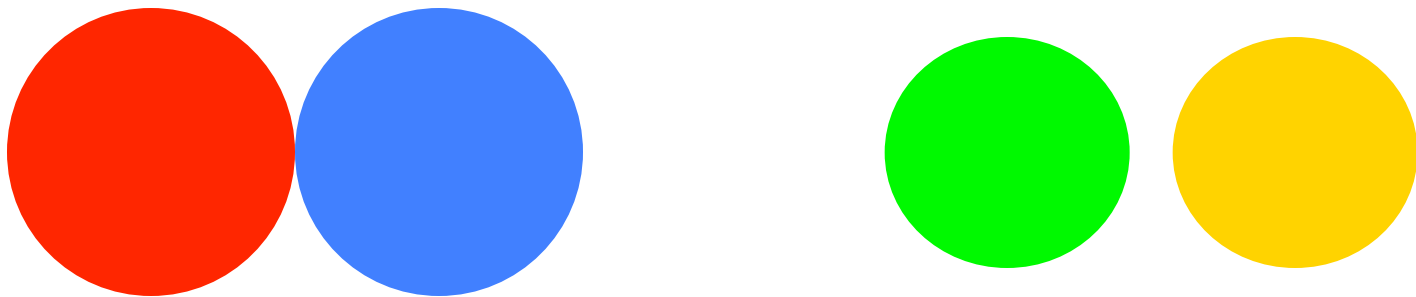


3 well-separated clusters

Types of Clusters: Center-Based

- Center-based

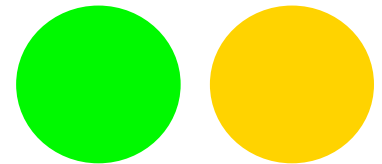
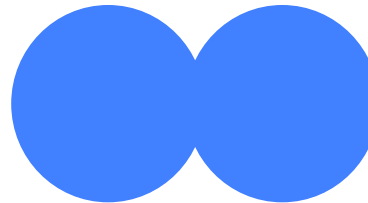
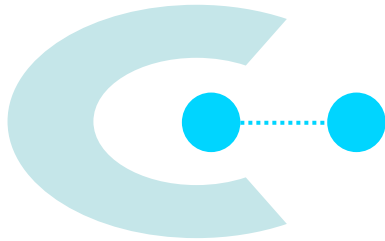
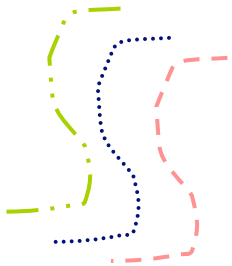
- A cluster is a set of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster
- The center of a cluster is often a **centroid**, the average of all the points in the cluster, or a **medoid**, the most “representative” point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



4 center-based clusters

Types of Clusters: Contiguity-Based

- **Contiguous Cluster (Nearest neighbor or Transitive)**
 - 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.

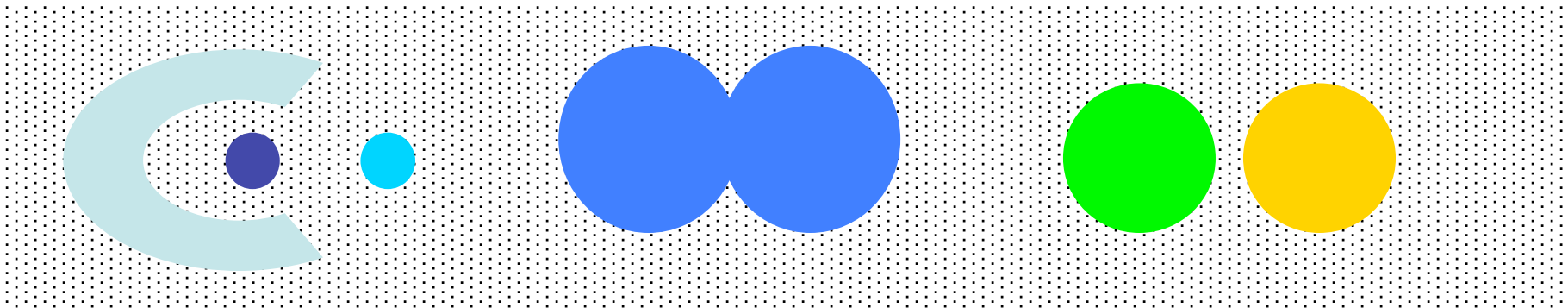


8 contiguous clusters

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.

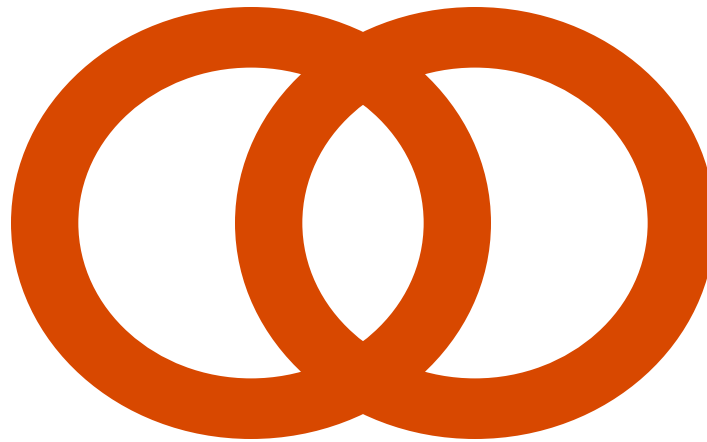


6 density-based clusters

Types of Clusters: Conceptual Clusters

- Shared Property or Conceptual Clusters
 - Finds clusters that share some common property or represent a particular concept.

.



2 Overlapping Circles

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.

Types of Clusters: Objective Function ...

- Map the clustering problem to a different domain and solve a related problem in that domain
 - Proximity matrix defines a weighted graph, where the nodes are the points being clustered, and the weighted edges represent the proximities between points
 - Clustering is equivalent to breaking the graph into connected components, one for each cluster.

Characteristics of the Input Data Are Important

- Type of proximity or density measure
 - This is a derived measure, but central to clustering
- Sparseness
 - Dictates type of similarity
 - Adds to efficiency
- Attribute type
 - Dictates type of similarity
- Type of Data
 - Dictates type of similarity
 - Other characteristics, e.g., autocorrelation
- Dimensionality
- Noise and Outliers
- Type of Distribution

Several Clustering Algorithms

- Partitioning Methods
 - K-means and k-medoids
- Hierarchical Methods
 - Agglomerative
- Density-based Methods
 - DBSCAN

K-means Clustering

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

K-means in Euclidean Space

-
- 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
-

In Euclidean Space, the centroid is the mean of all data points within the cluster. It has been mathematically proven that the centroid that minimizes the SSE (sum of the squared error) is the mean.

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
 - can show that m_i corresponds to the center (mean) of the cluster
- Given two clusters, we can choose 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

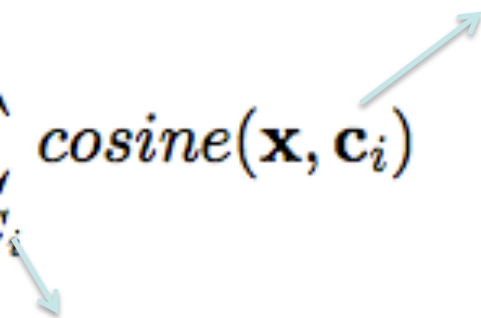
K-means in other spaces

Proximity Function	Centroid	Objective Function
Manhattan (L_1)	median	Minimize sum of the L_1 distance of an object to its cluster centroid
Squared Euclidean (L_2^2)	mean	Minimize sum of the squared L_2 distance of an object to its cluster centroid
cosine	mean	Maximize sum of the cosine similarity of an object to its cluster centroid
Bregman divergence	mean	Minimize sum of the Bregman divergence of an object to its cluster centroid

Document Data: cosine similarity

	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

Cluster
center

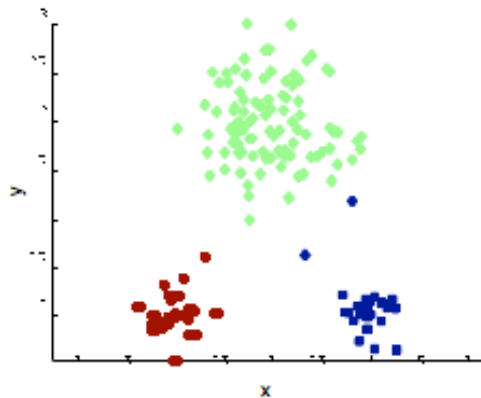
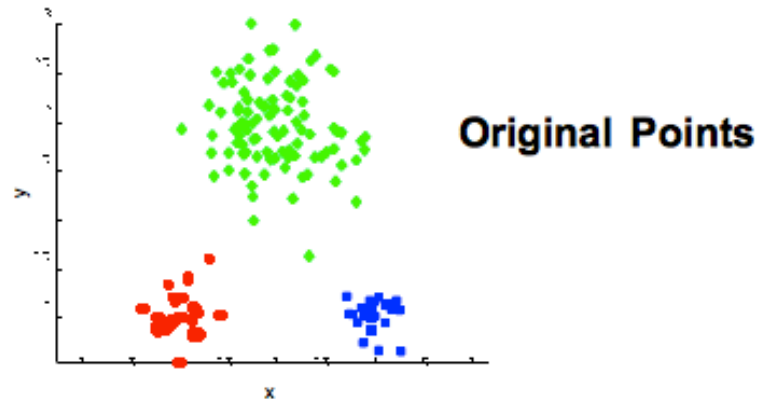
$$\text{Total Cohesion} = \sum_{i=1}^K \sum_{\mathbf{x} \in C_i} \text{cosine}(\mathbf{x}, \mathbf{c}_i)$$


cluster

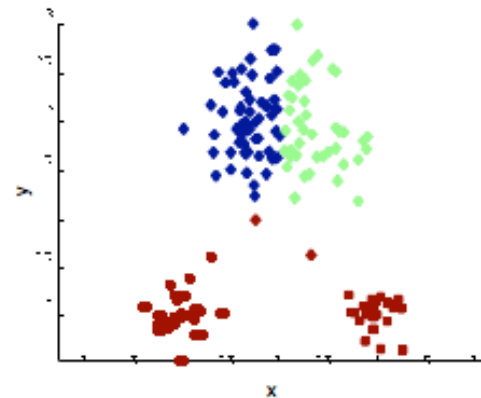
K-means Clustering - More Details

- 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

Two different K-means Clusterings



Optimal Clustering



Sub-optimal Clustering

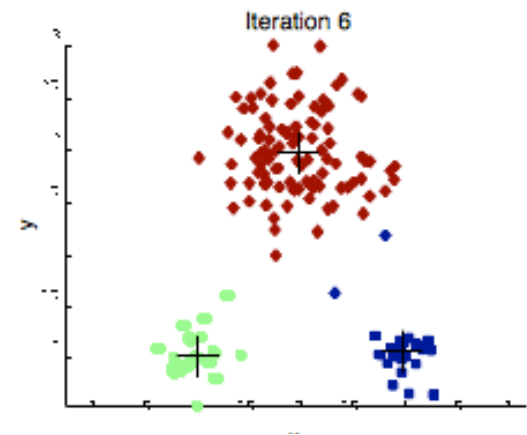
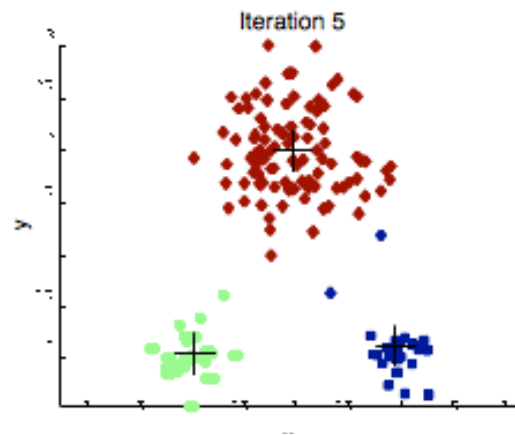
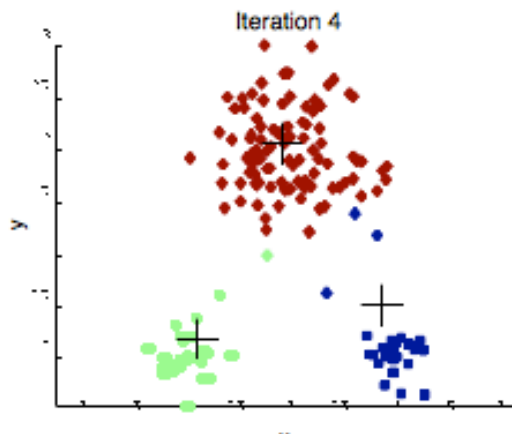
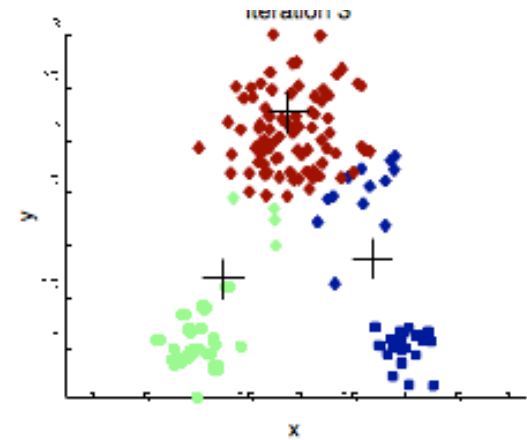
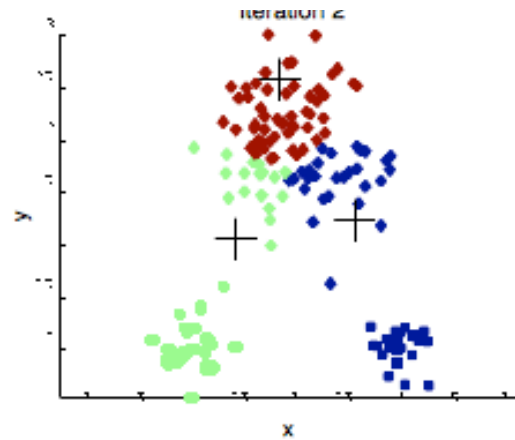
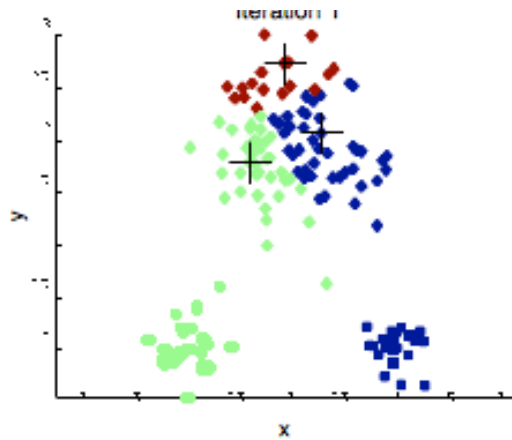
Interactive Demo

- http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/AppletKM.html

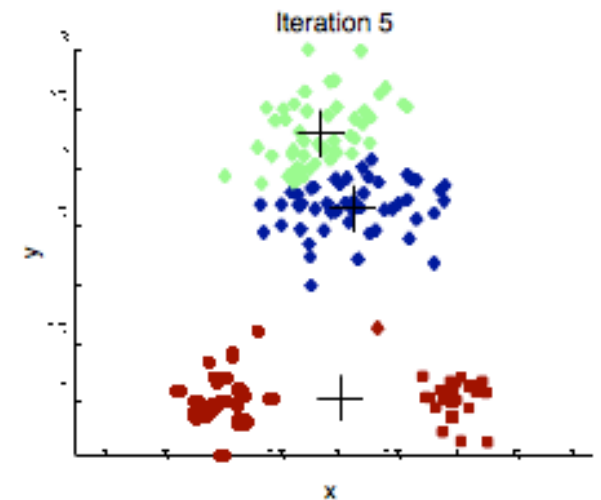
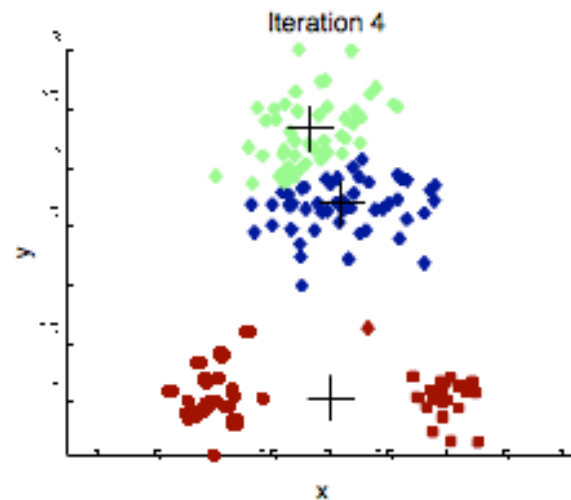
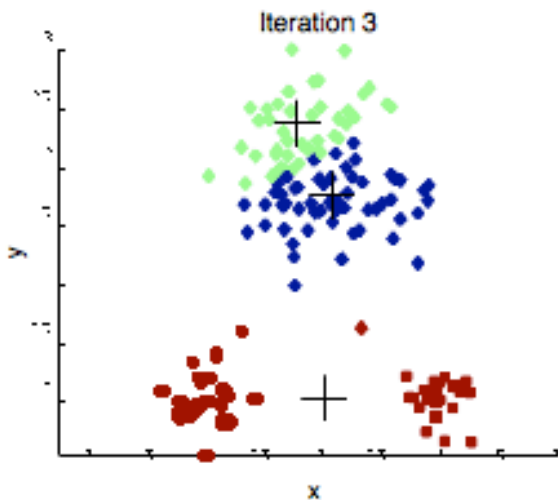
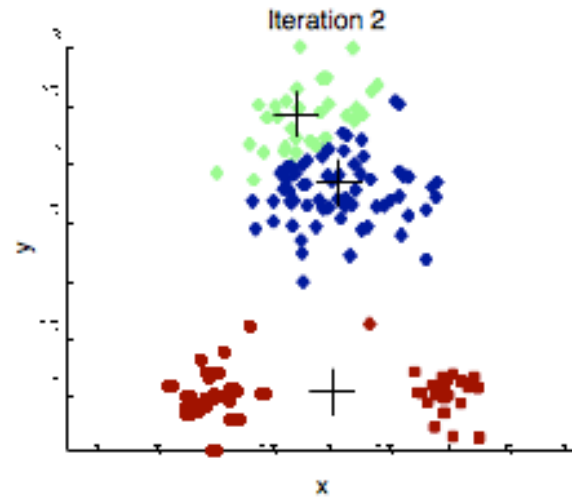
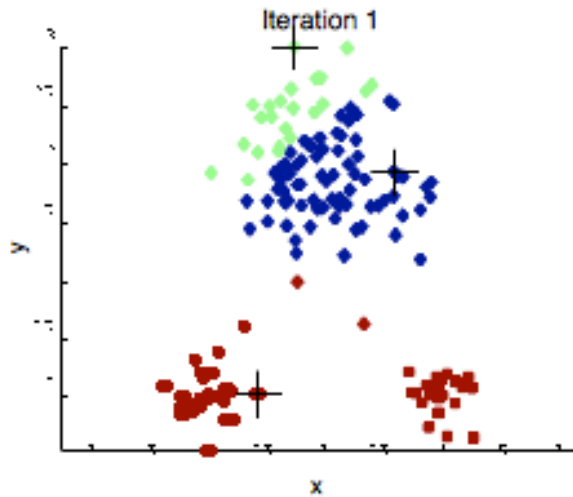
Choosing the number of Clusters

- K-means method expects number of clusters k to be specified as part of the input
- How to choose appropriate k ?
- Options
 - Start with $k = 1$ and work up to small number
 - Select the result with lowest squared error
 - Warning: Result with each instance in a different cluster on its own would be the best
 - Create two initial clusters and split each cluster independently
 - When splitting a cluster create two new 'seeds'
 - One seed at a point one standard deviation away from the cluster center and the second one standard deviation in the opposite direction
 - Apply k-means to the points in the cluster with the two new seeds

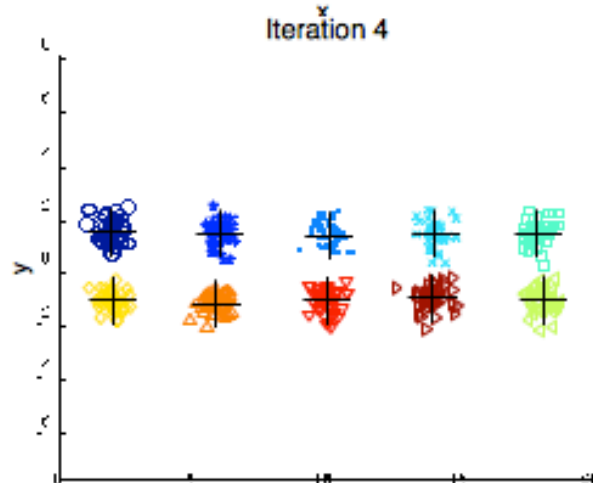
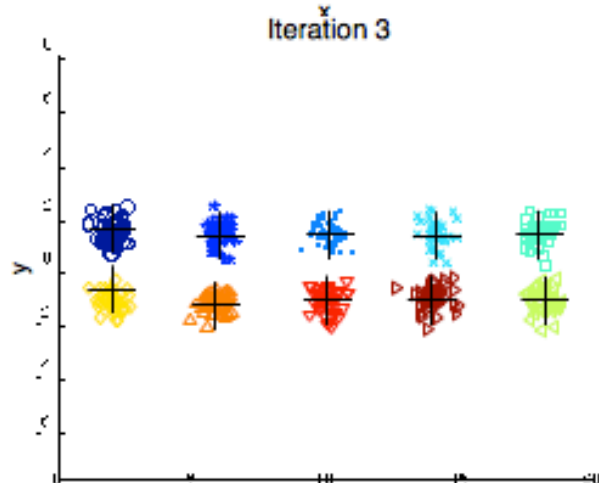
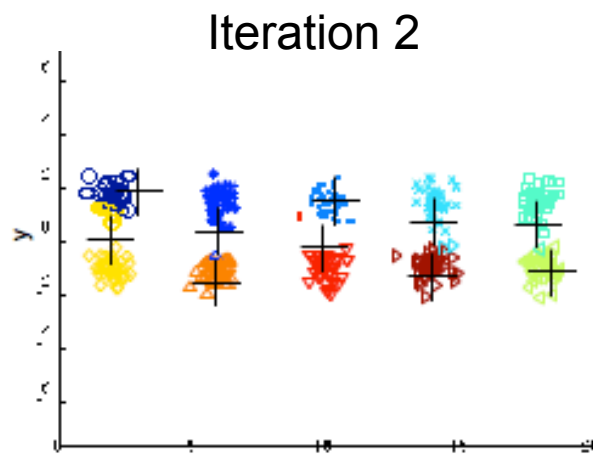
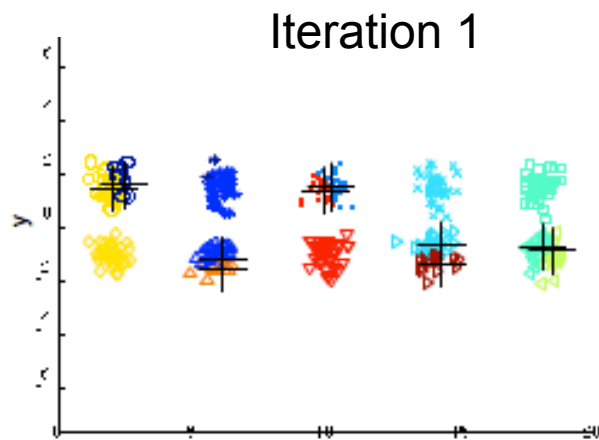
Importance of Choosing Initial Centroids



Importance of Choosing Initial Centroids

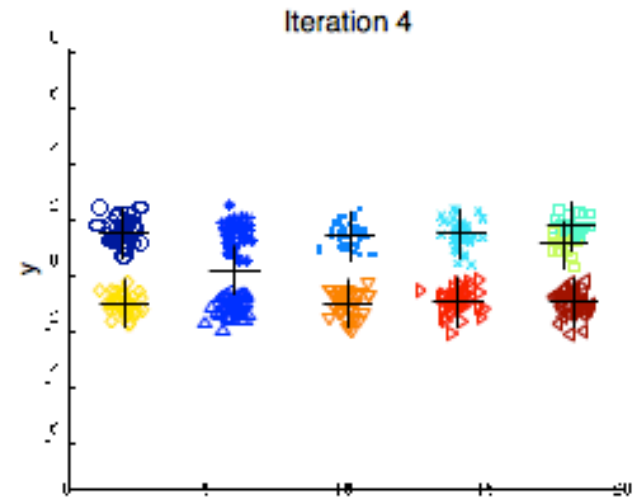
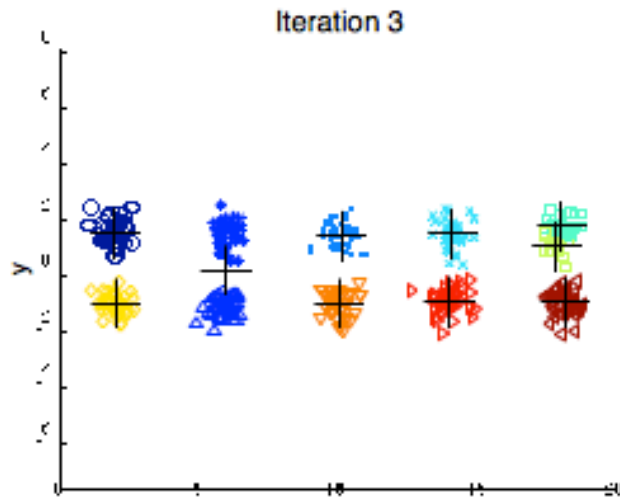
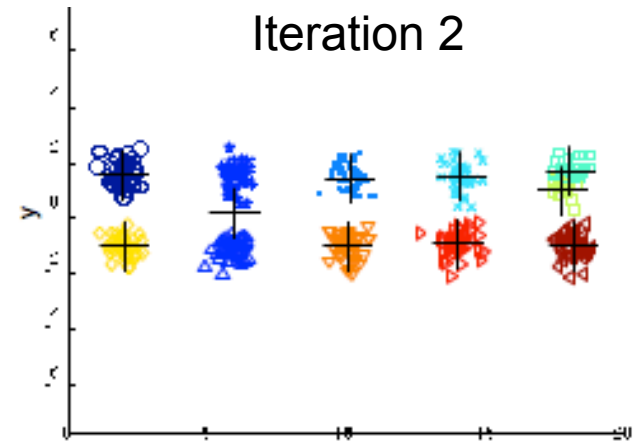
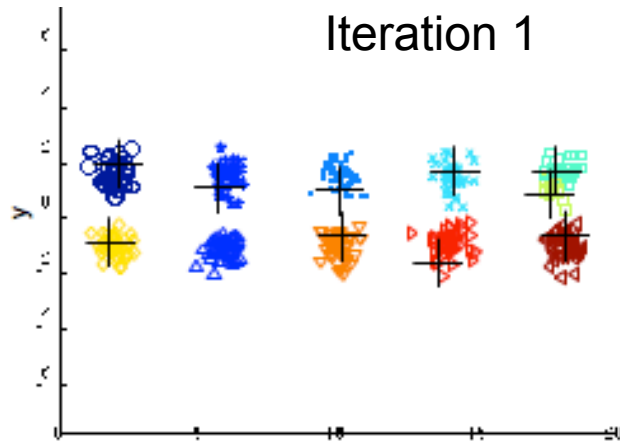


10 Clusters Example



Starting with two initial centroids in one cluster of each pair of clusters 33

10 Clusters Example



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

Choosing Initial Centroids

- Randomly chosen centroids may produce poor quality clusters
- How to choose initial centroids?
- Options
 - Take a sample of data, apply hierarchical clustering and extract k cluster centroids
 - Works well - small sample sizes are desired
 - Take well separated k points
 - Take a sample of data and select the first centroid as the centroid of the sample
 - Each of the subsequent centroid is chosen to be a point farthest from any of the already selected centroids

Solutions to Initial Centroids Problem

- Multiple runs
 - Helps, but probability is not on your side
- Sample 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
- Postprocessing
- Bisecting K-means
 - Not as susceptible to initialization issues

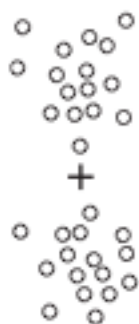
Bisecting K-means

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

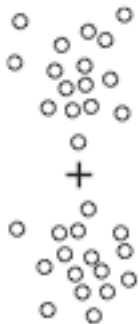
Algorithm 8.2 Bisecting K-means algorithm.

```
1: Initialize the list of clusters to contain the cluster consisting of all points.
2: repeat
3:   Remove a cluster from the list of clusters.
4:   {Perform several “trial” bisections of the chosen cluster.}
5:   for  $i = 1$  to number of trials do
6:     Bisect the selected cluster using basic K-means.
7:   end for
8:   Select the two clusters from the bisection with the lowest total SSE.
9:   Add these two clusters to the list of clusters.
10: until Until the list of clusters contains  $K$  clusters.
```

Examples of Bisecting K-means



(a) Iteration 1.



(b) Iteration 2.



(c) Iteration 3.

Summary

- Basic Knowledge and concepts of Clustering Analysis
 - Types of clustering
 - Types of cluster
- K-means
- Bisecting K-means

Next Lecture

- Issues & Limitations of K-means
- EM algorithm , Cobweb/Classit
- Hierarchical clustering
- Read Chapter 8.2 and 8.3 of the Kumar Book.
 - Chapter 8 is Online free
 - Don't need to read Chap 8.2.6 (unless you want to) and other part with heavy math
- Optional Reading: Chapter 9.2.2

Acknowledgement

- Some of the slides are based on or taken from the course slides provided by
 - Tan, Steinbach and Kumar (Introduction to Data Mining)
- Some pictures are taken from various online resources.