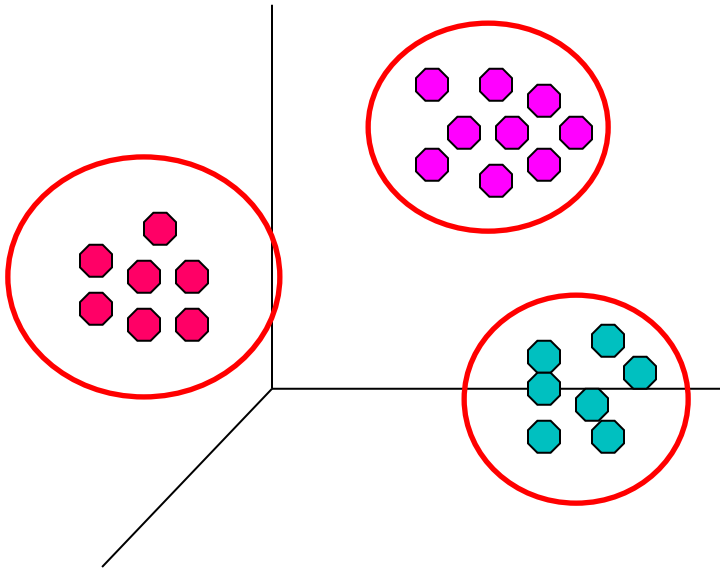
Clustering

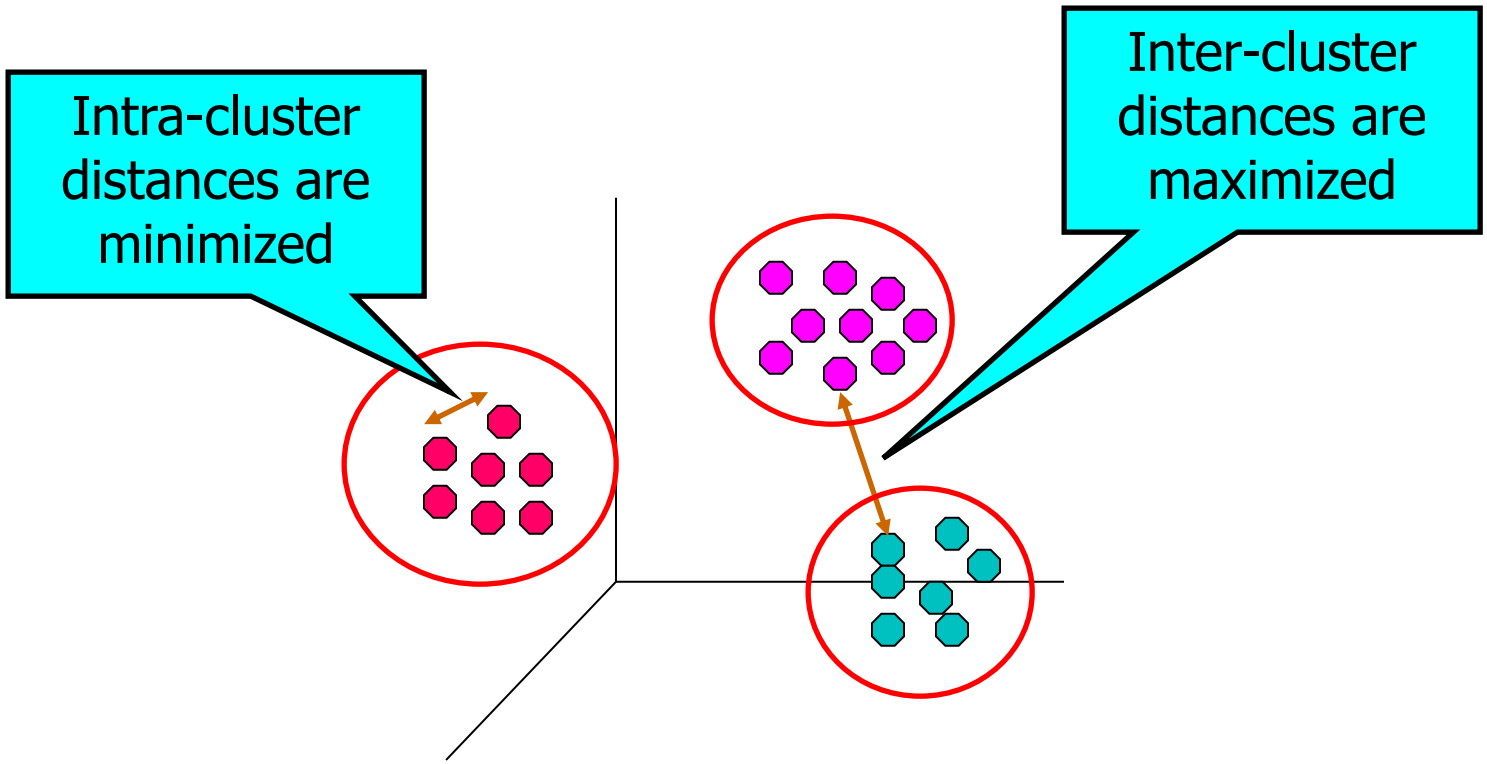
Unsupervised Learning

- With supervised learning, the instances are labelled
- With unsupervised learning, the instances are not
- What can we do with unlabelled data?
  - Clustering



What is Clustering?

- 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

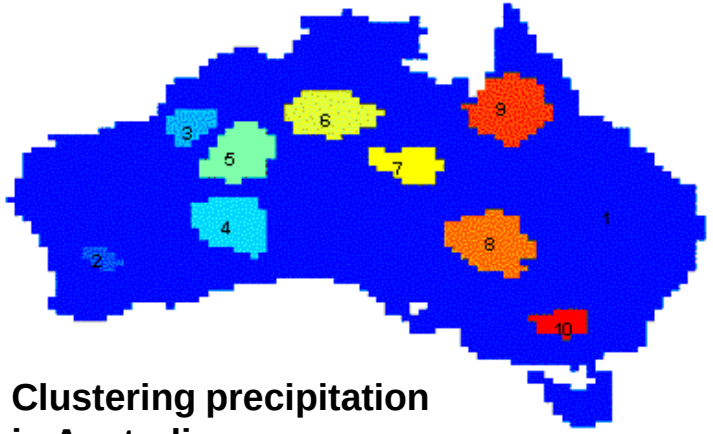


Applications of Cluster Analysis

- **Understanding**
  - Group related documents for browsing, 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

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



Clustering precipitation in Australia

Notion of a Cluster can be Ambiguous

How many clusters?

Six Clusters

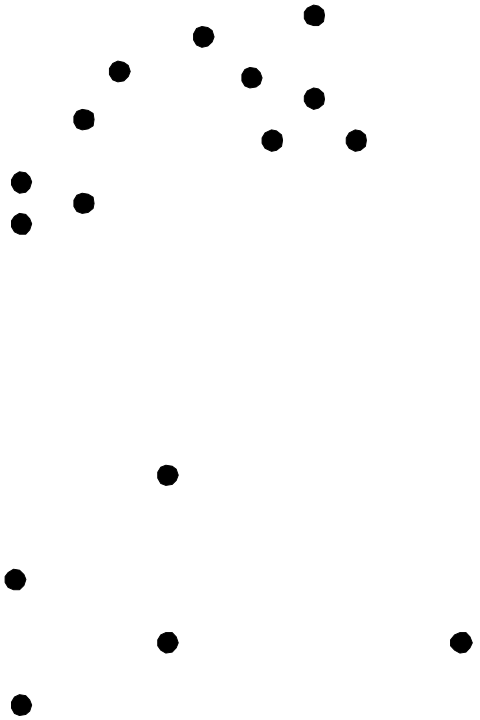
Two Clusters

Four Clusters

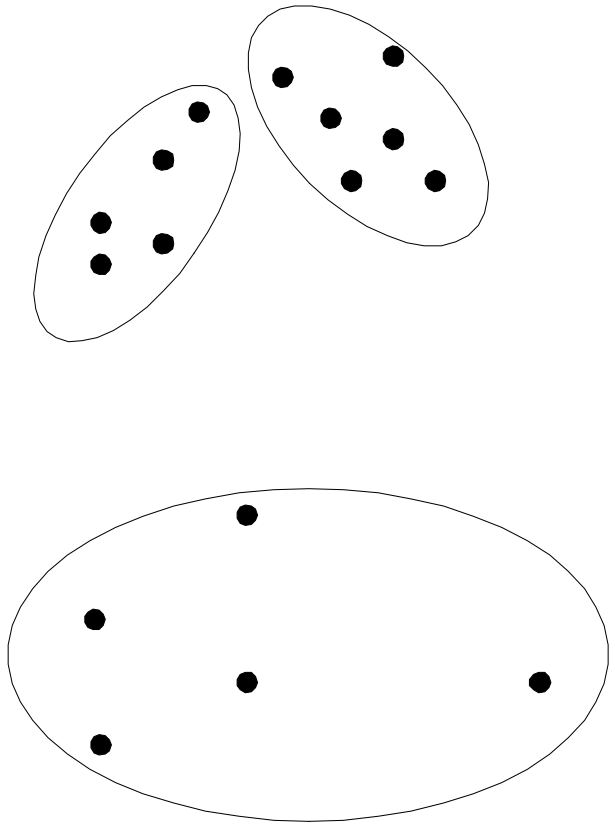
Types of Clusterings

- 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 organised as a hierarchical tree

Partitional Clustering

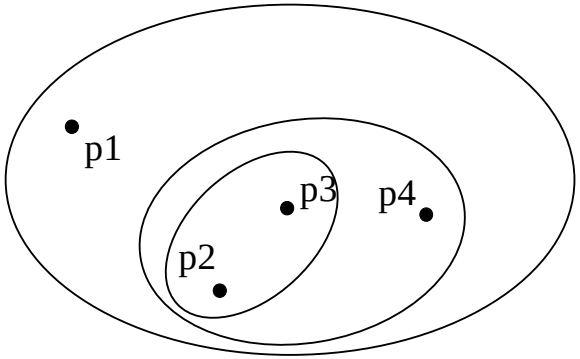


Original Points

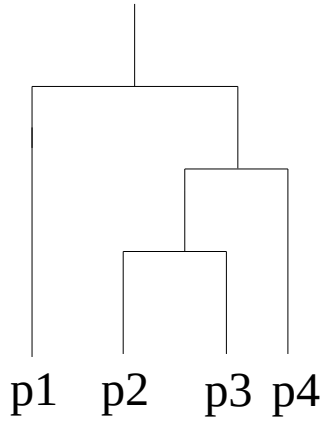


A Partitional Clustering

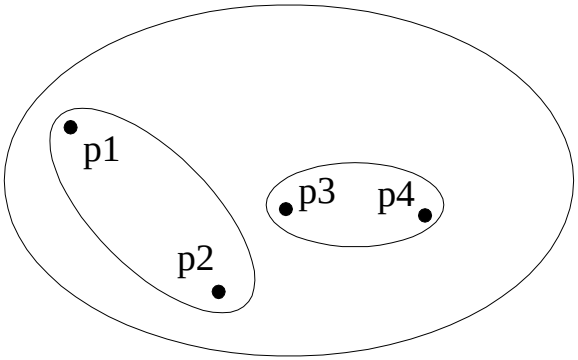
Hierarchical Clustering



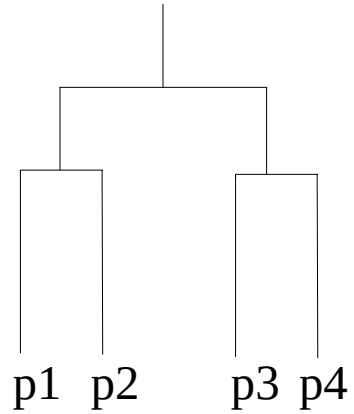
Traditional Hierarchical Clustering



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.
  - Allow ‘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

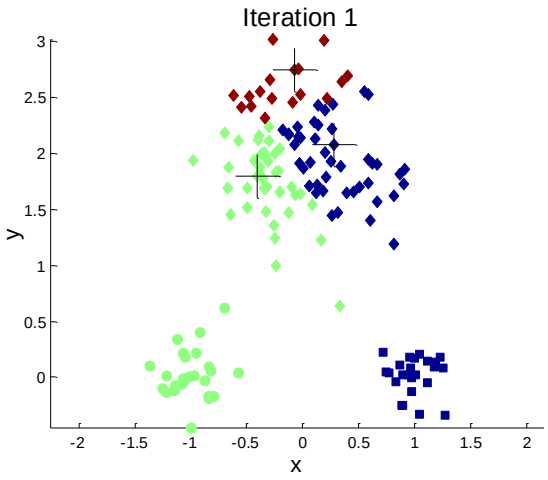
K-means Clustering

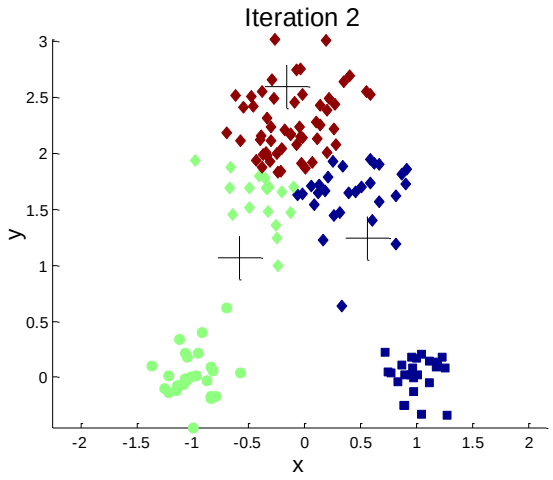
- Partitional clustering approach
- Number of clusters, K, is given
- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid

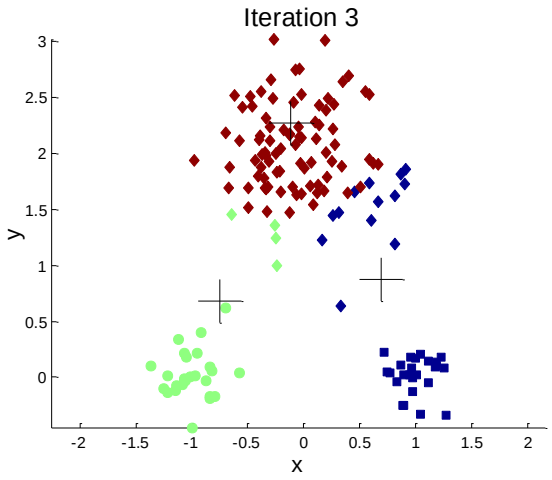
Algorithm Basic K-means algorithm.

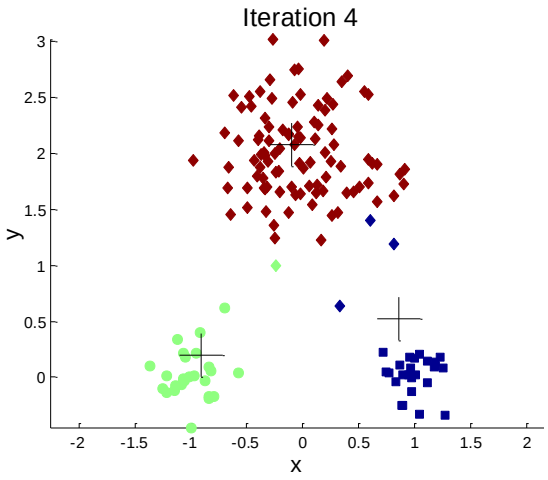
1: Select K points as initial centroids.
2: **repeat**
3: Form K clusters by assigning each point to its closest centroid.
4: Recompute the centroid of each cluster.
5: **until** Centroids do not change.

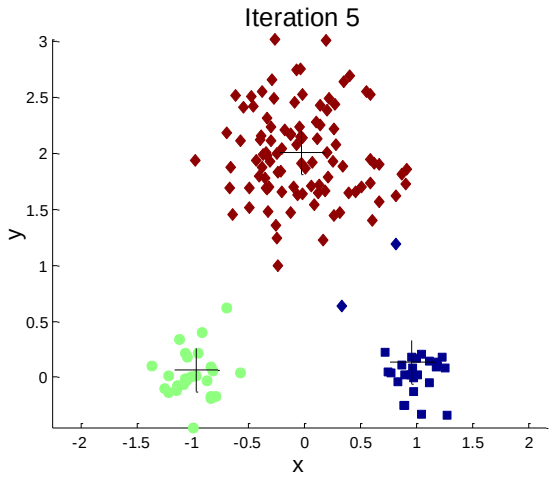
Example of K-means Clustering

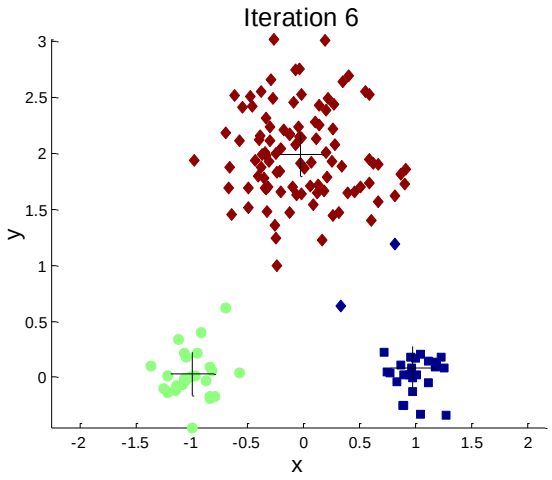












K-means Clustering

- Initial centroids are often chosen randomly.
  - Clusters produced depend on initial centroids chosen.
- The centroid is the mean of the points in the cluster.
- ‘Closeness’ is measured by Euclidean distance, cosine similarity, 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’

K-means Clustering

- Time complexity: $O(nKdI)$
  - n: number of data points
  - K: number of clusters
  - d: number of attributes
  - I: number of iterations
- Space complexity: $O((n + K)d)$

Evaluating K-means Clusters

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

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

  - K is the number of clusters, $\boldsymbol{x}$ is a data point in cluster C_i and $\boldsymbol{c}_i$ is the centroid (mean) of cluster C_i
  - Given two sets of clusters, we prefer the one with the smaller SSE
  - K-means is guaranteed to find a local minimum, but not necessarily the global minimum
  - One easy way to reduce SSE is to increase K
    - A good clustering with smaller K can have a lower SSE than a poor clustering with higher K

Derivation of K-means (Minimizing SSE)

$$SSE = \sum_{i=1}^{K} \sum_{\boldsymbol{x} \in C_i} (\boldsymbol{c}_i - \boldsymbol{x})^2 = \sum_{i=1}^{K} \sum_{\boldsymbol{x} \in C_i} \sum_{j=1}^{d} (c_{i,j} - x_j)^2$$

$$\frac{\partial}{\partial c_{k,j}} SSE = \sum_{x_j \in C_k} 2(c_{k,j} - x_j) = 0$$

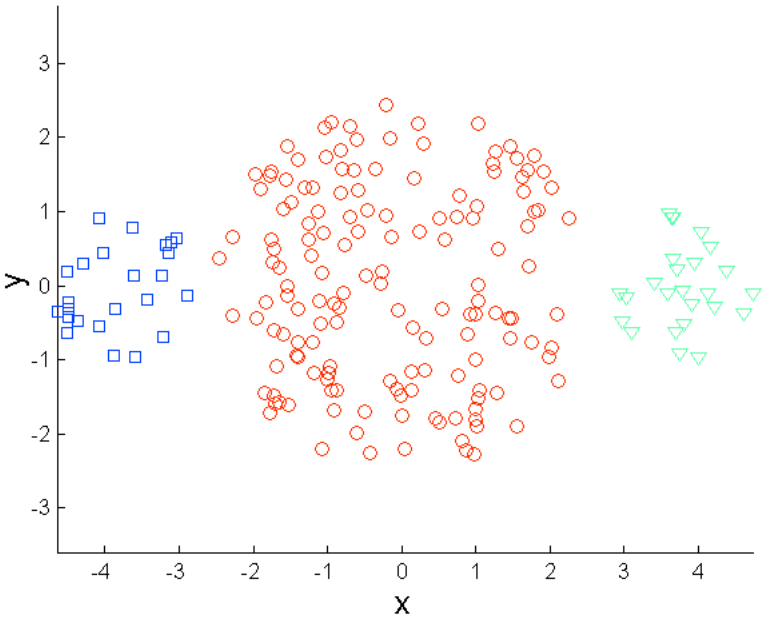
$$c_{k,j} = \frac{1}{n_k} \sum_{x_j \in C_k} x_j$$

$$\boldsymbol{c}_k = \frac{1}{n_k} \sum_{\boldsymbol{x} \in C_k} \boldsymbol{x}$$

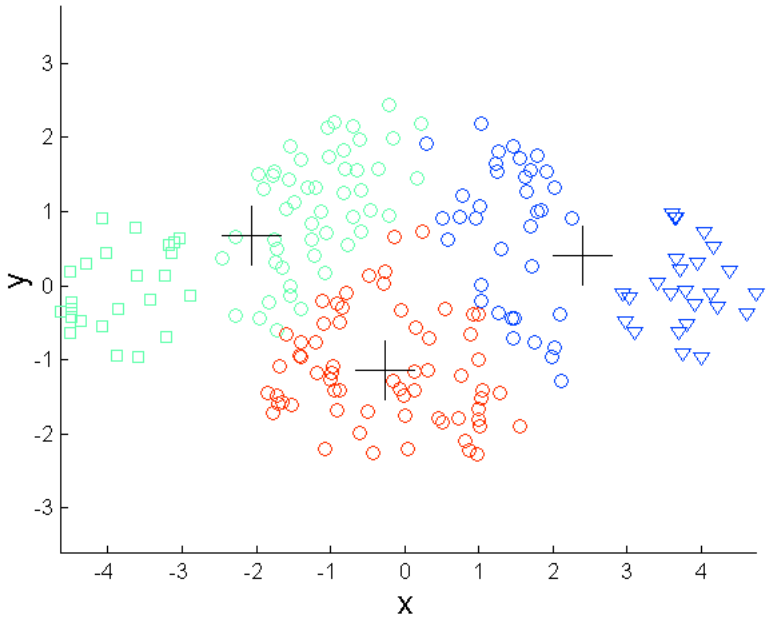
Limitations of K-means

- K-means has problems when clusters are of differing
  - Sizes
  - Densities
  - Non-globular (non-spherical) shapes
- K-means has problems when the data contains outliers.

Limitations of K-means: Differing Sizes

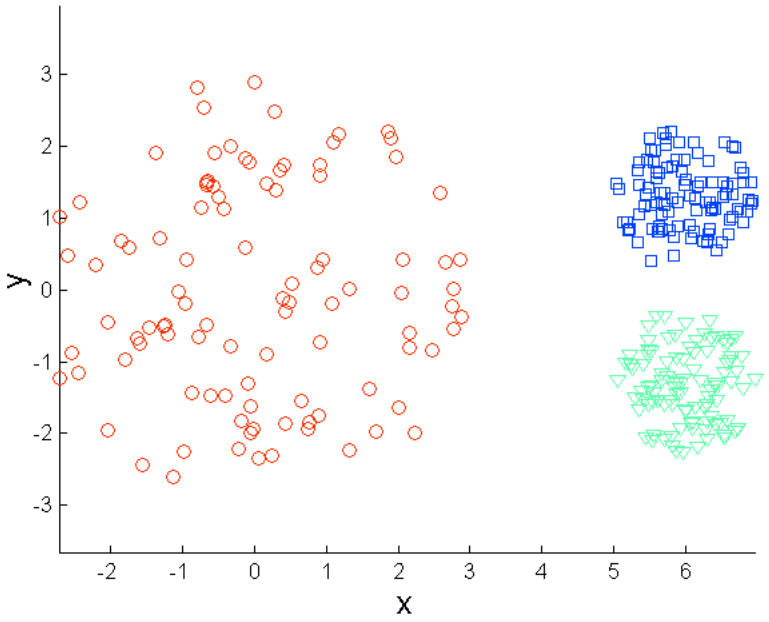


Original Points

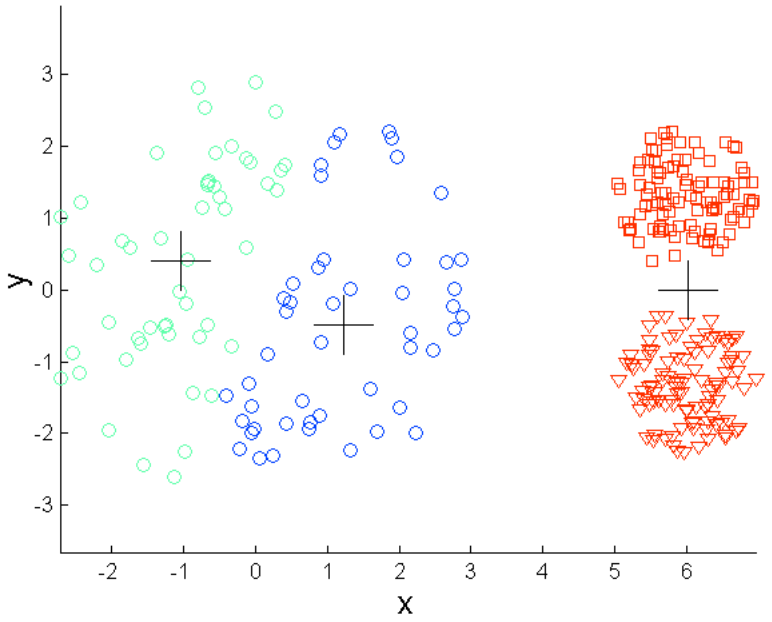


K-means (3 Clusters)

Limitations of K-means: Differing Density

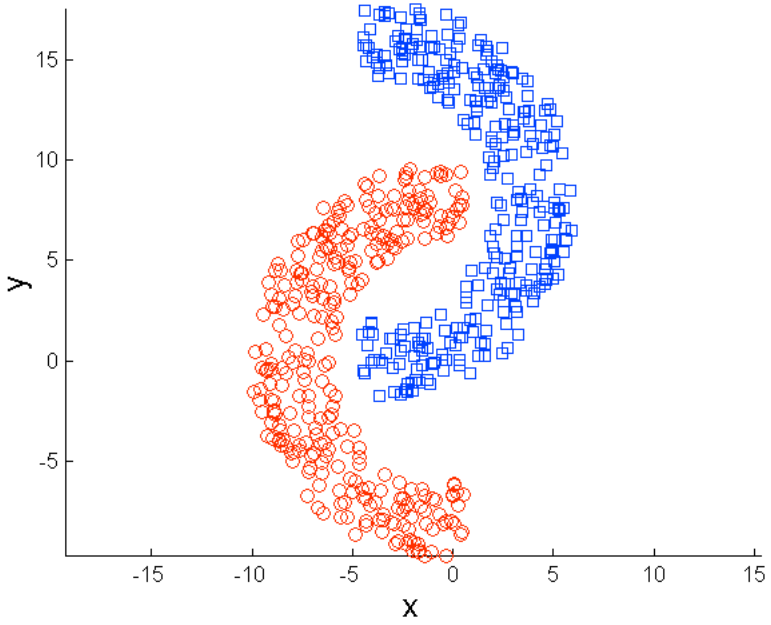


Original Points

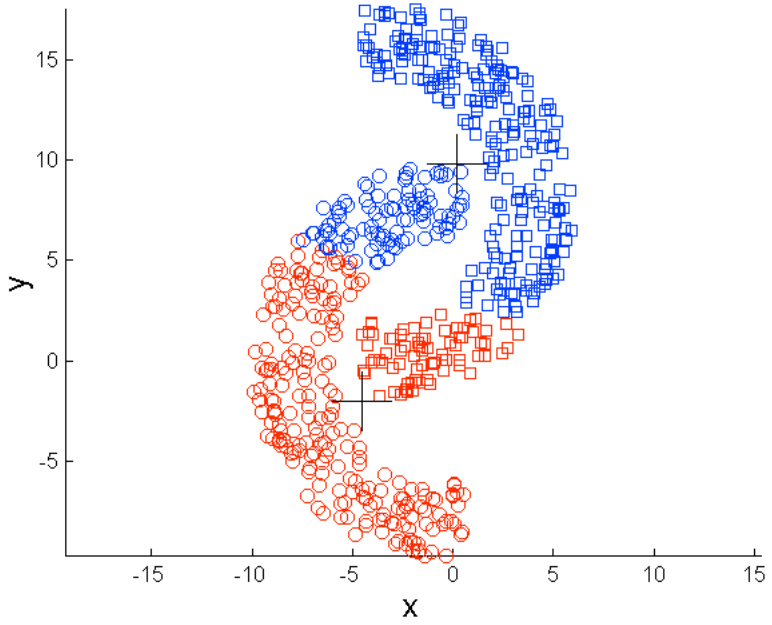


K-means (3 Clusters)

Limitations of K-means: Non-globular Shapes

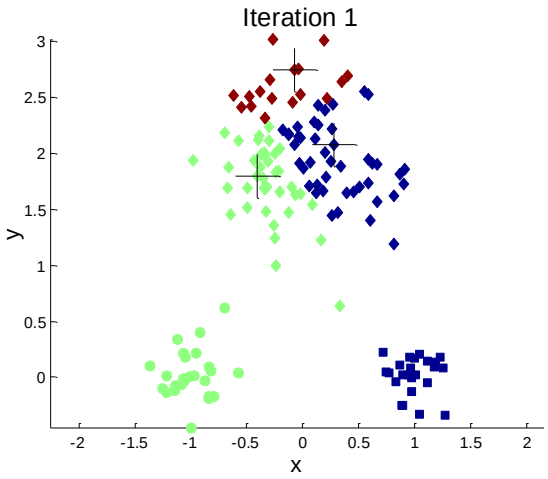


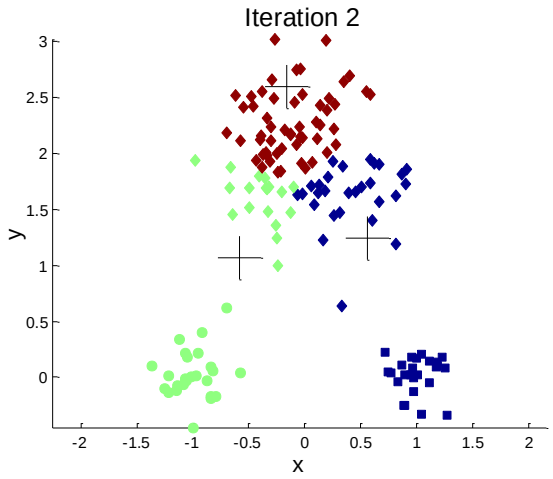
Original Points

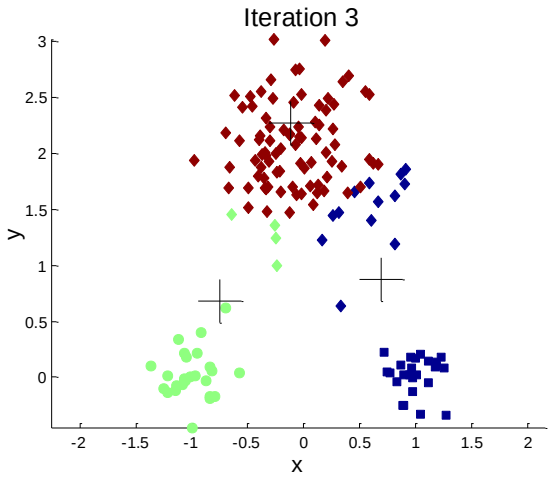


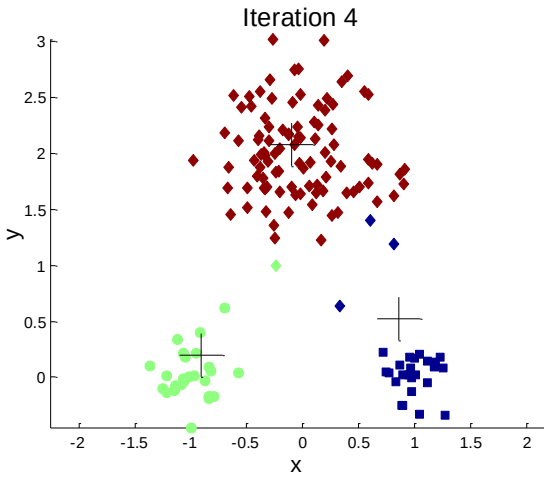
K-means (2 Clusters)

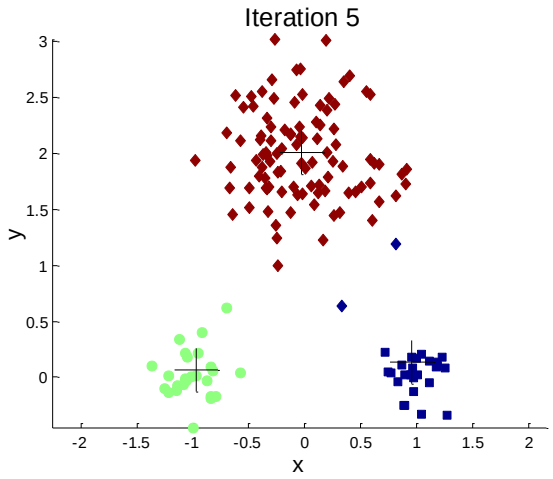
Importance of Choosing Initial Centroids

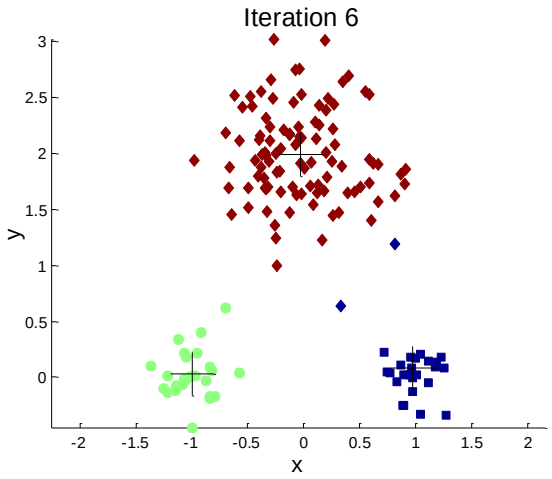




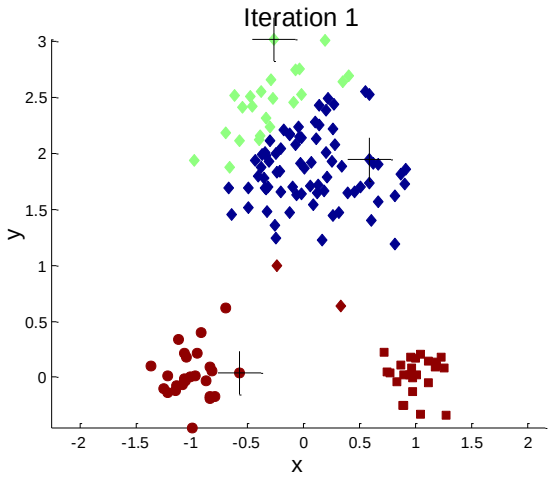


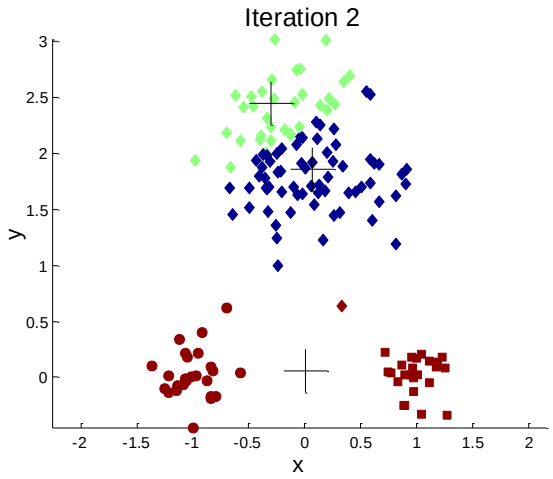


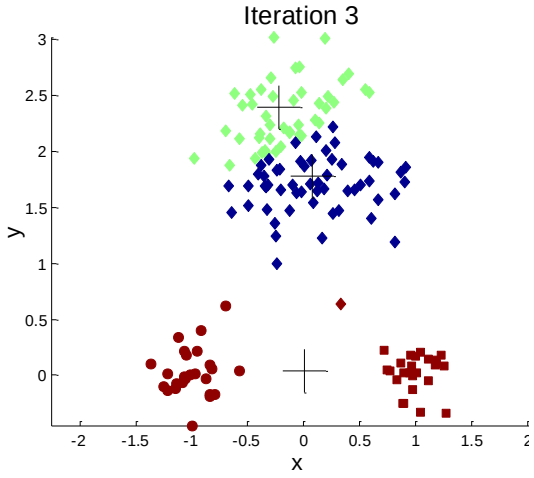


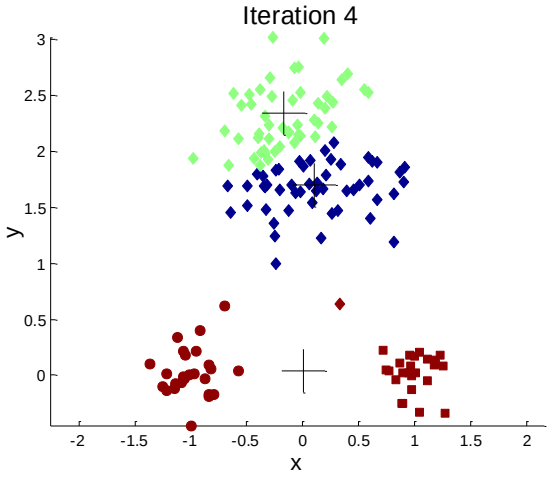


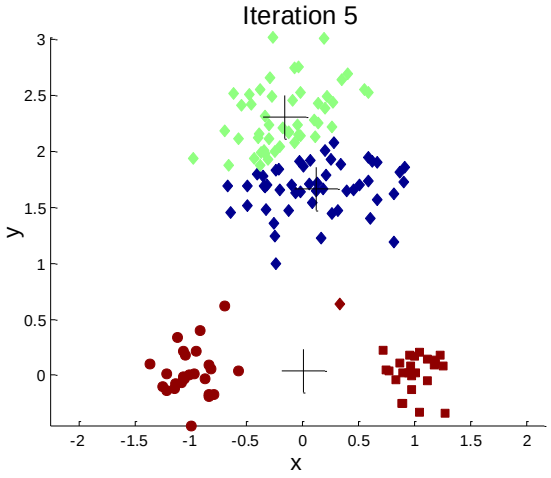
Importance of Choosing Initial Centroids



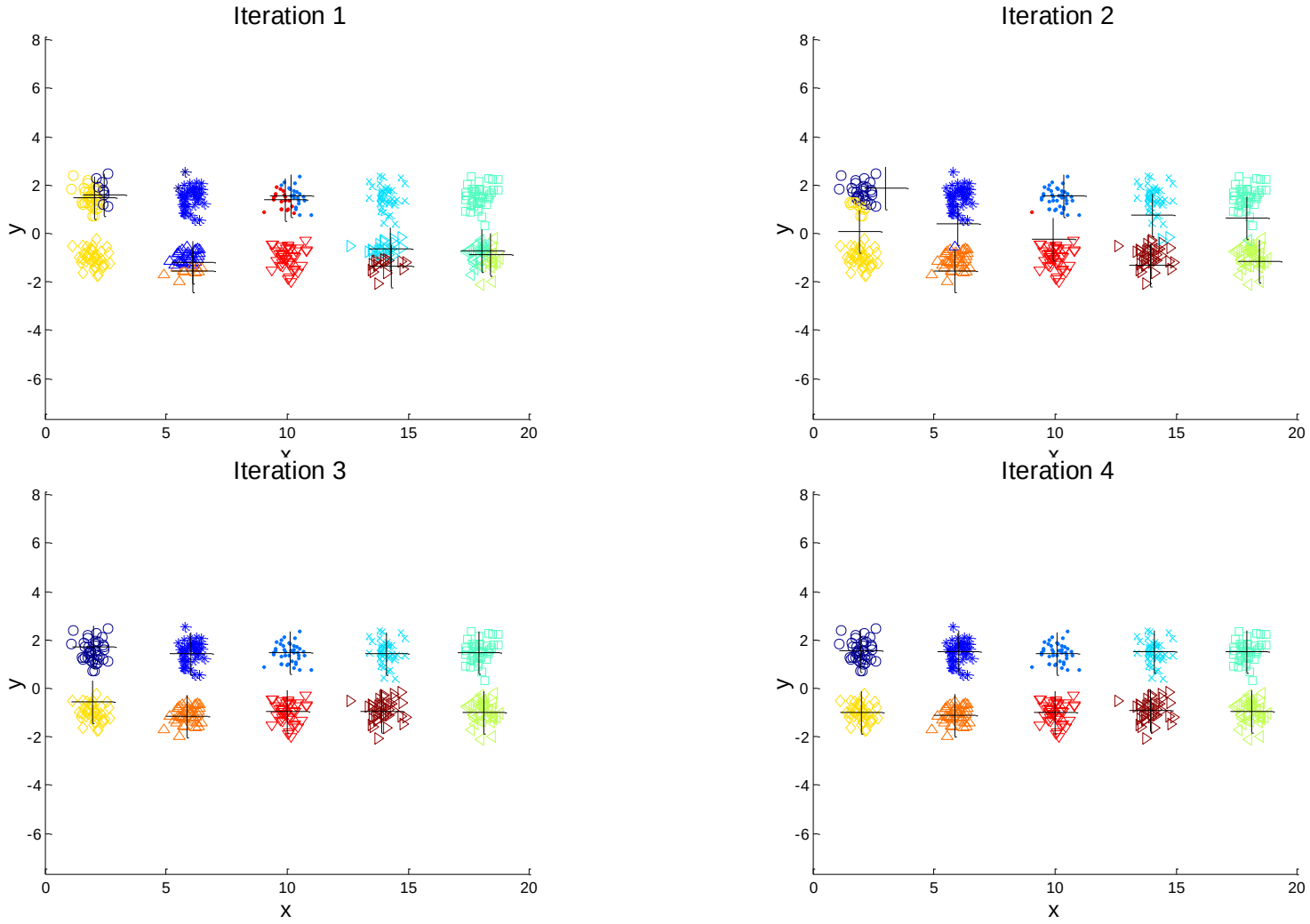






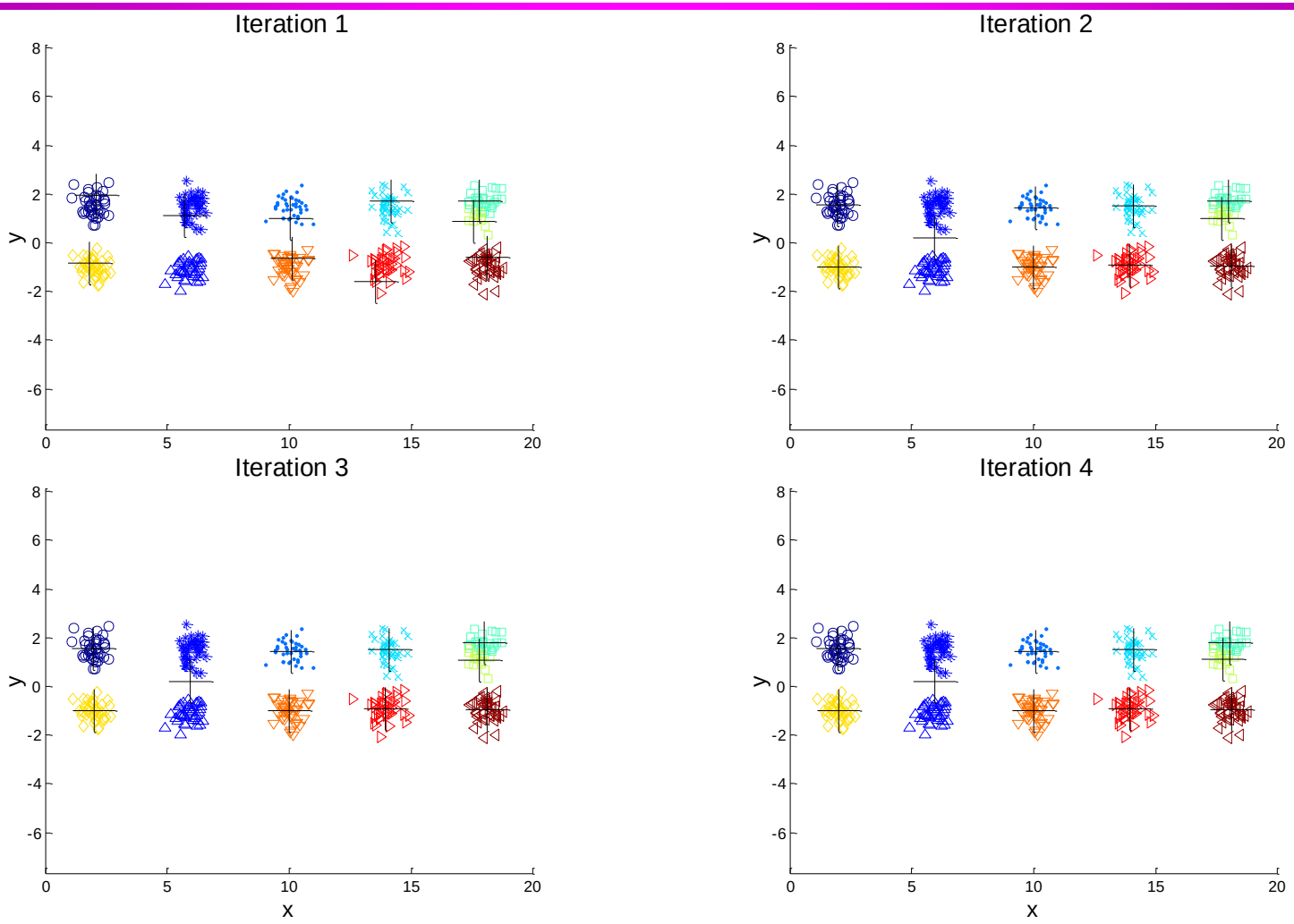


10 Clusters Example



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

10 Clusters Example



Starting with some pair of clusters having three initial centroids, while another pair has only one.

Solutions to Initial Centroids Problem

- Multiple runs with different initial centroids
- K-means++
  - An algorithm to select the initial centroids, followed by running regular K-means
  - Guaranteed to find a clustering that is optimal to within a factor of $O(\log K)$
- Bisecting K-means
  - Less susceptible to initialisation problems

K-means++ Initialisation

- This approach can be slower than random initialisation, but very consistently produces better results in terms of SSE
- To select a set of initial centroids, C_j, perform the following:

1. Select an initial point at random to be the first centroid C_1
2. For j = 1 to K – 1
3. For each of the data points x_i, find the minimum squared distance to the currently selected centroids, $C_1, \ldots, C_j, 1 \leq j \leq \mathrm{K}-1$, i.e., $\min_j d^2(C_j, x_i)$
4. Select x_i as the new centroid C_{j+1}, with probability $$\frac{\min_j d^2(C_j, x_i)}{\sum_i \min_j d^2(C_j, x_i)}$$
5. End for

Bisecting K-means

- A variant of K-means that can produce a partitional or a hierarchical clustering
- Bisect a cluster (split it into two) at each step
- Choice of cluster to bisect:
  - The cluster with the largest number of data points
  - The cluster with the largest SSE

Bisecting K-means

Algorithm 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** The list of clusters contains K clusters.

Pre-processing and Post-processing

- Pre-processing
  - Eliminate outliers
- Post-processing
  - Eliminate small clusters that may represent outliers
  - Merge clusters that are ‘close’ and that have relatively low SSE
  - Split ‘loose’ clusters, i.e., clusters with relatively high SSE

Questions?