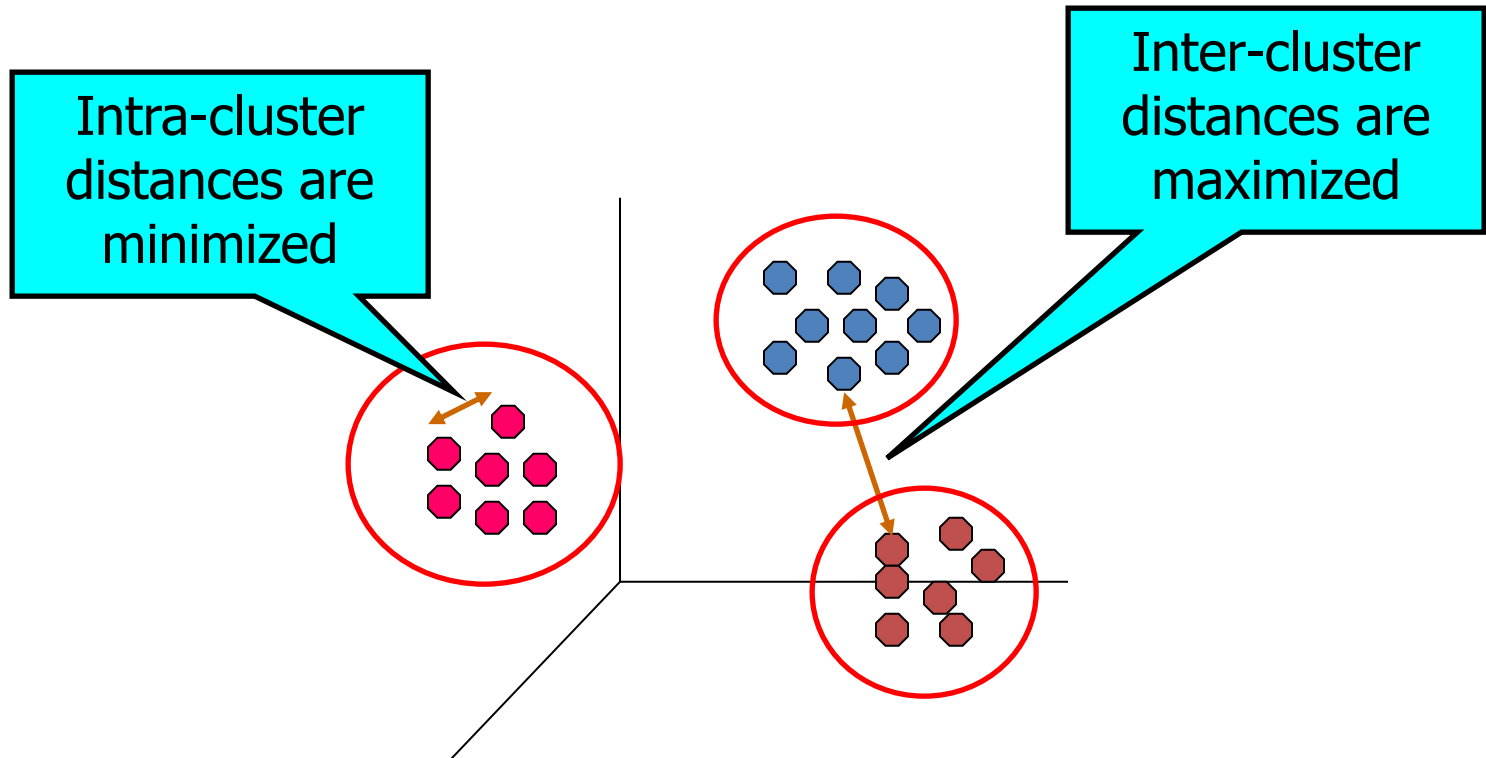


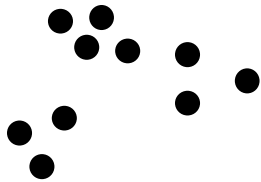
Cluster Analysis

What is Cluster Analysis?

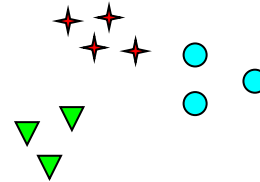
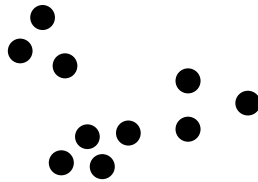
- Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



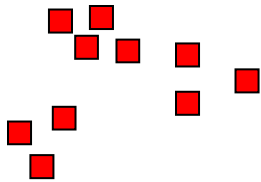
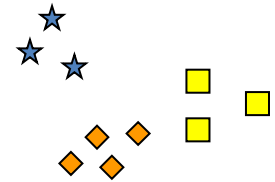
Notion of a Cluster can be Ambiguous



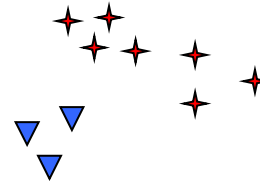
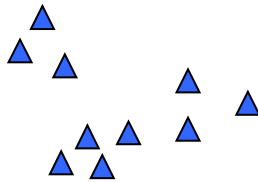
How many clusters?



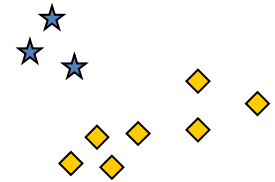
Six Clusters



Two Clusters



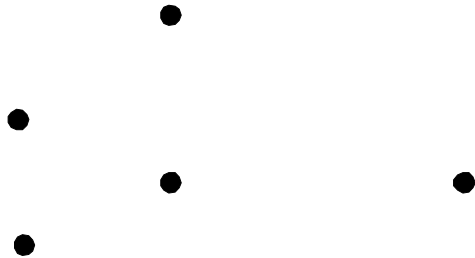
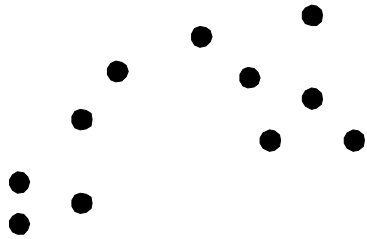
Four Clusters



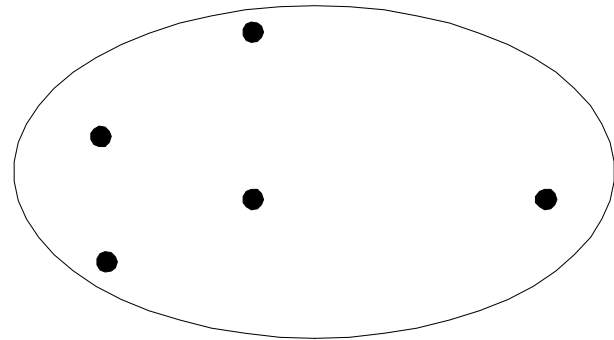
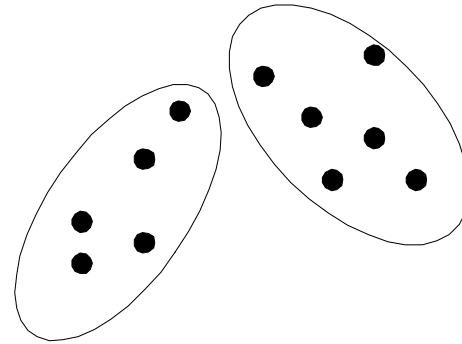
Types of Clusterings

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

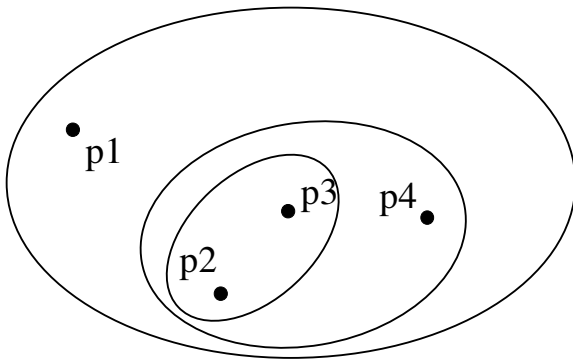


Original Points

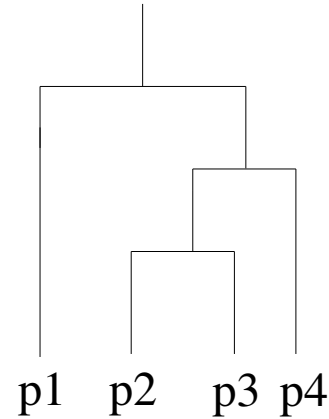


A Partitional Clustering

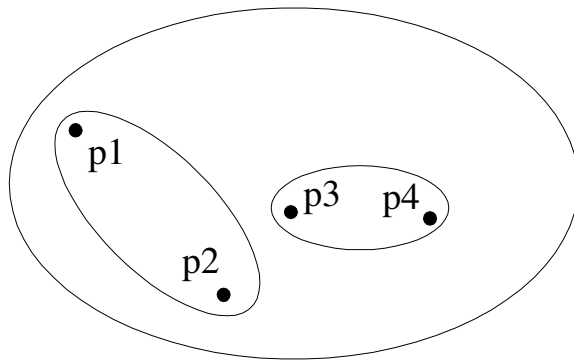
Hierarchical Clustering



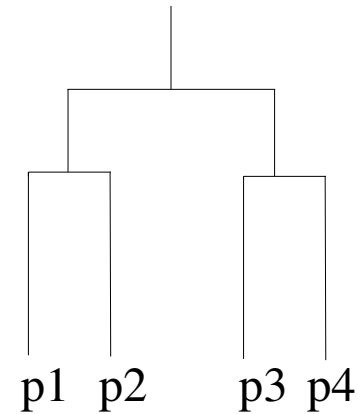
Traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Hierarchical Clustering



Non-traditional Dendrogram

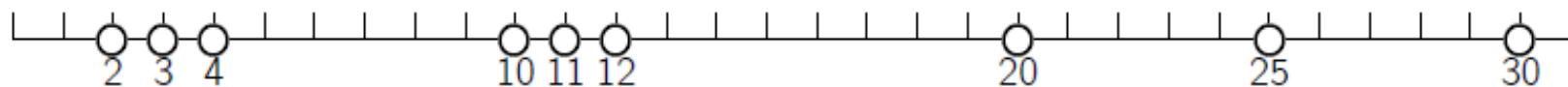
Clustering Algorithms

- Partitional Clustering
 - K-means and its variants
- Hierarchical clustering
 - Agglomerative(Bottom-up) clustering
 - Top-Down clustering

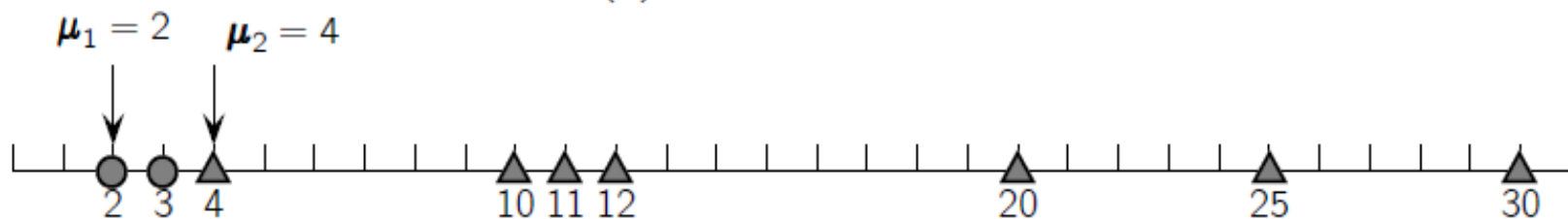
K-means Clustering

- 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

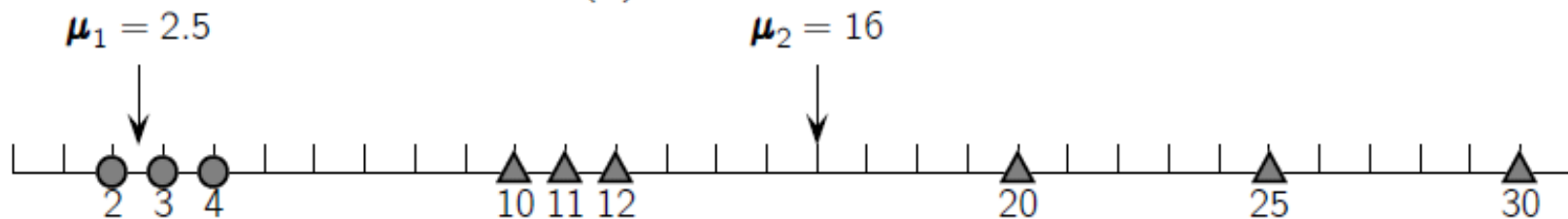
-
- 1: Select K points as the initial centroids.
 - 2: **repeat**
 - 3: Form K clusters by assigning all points to the closest centroid.
 - 4: Recompute the centroid of each cluster.
 - 5: **until** The centroids don't change
-



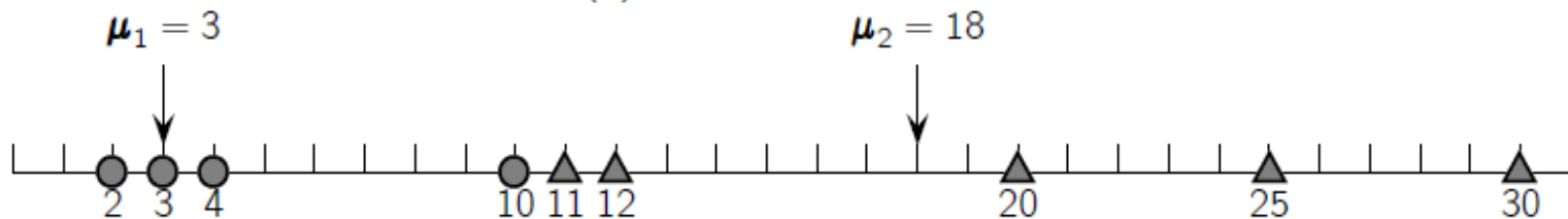
(a) Initial dataset



(b) Iteration: $t = 1$



(c) Iteration: $t = 2$



(d) Iteration: $t = 3$



(e) Iteration: $t = 4$

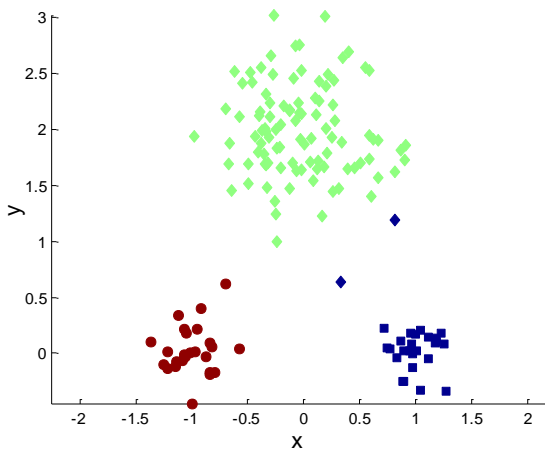
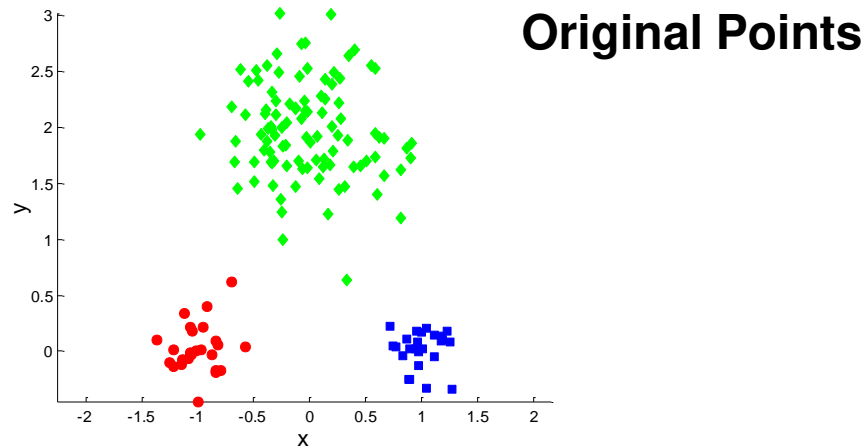


(f) Iteration: $t = 5$ (converged)

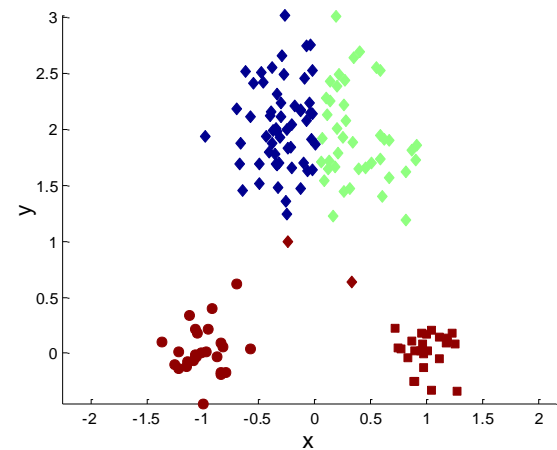
Issues and Limitations for K-means

- How to choose initial centers?
- How to choose K?
- How to handle Outliers?
- Clusters different in
 - Shape
 - Density
 - Size

Two different K-means Clusterings

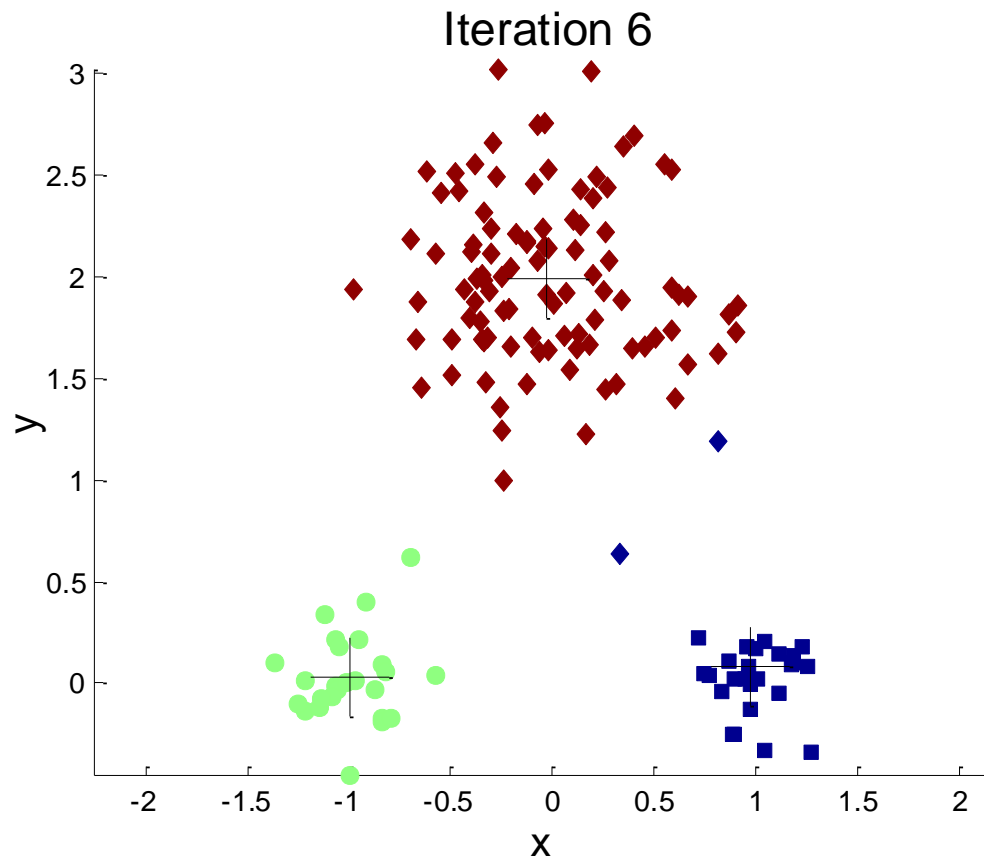


Optimal Clustering

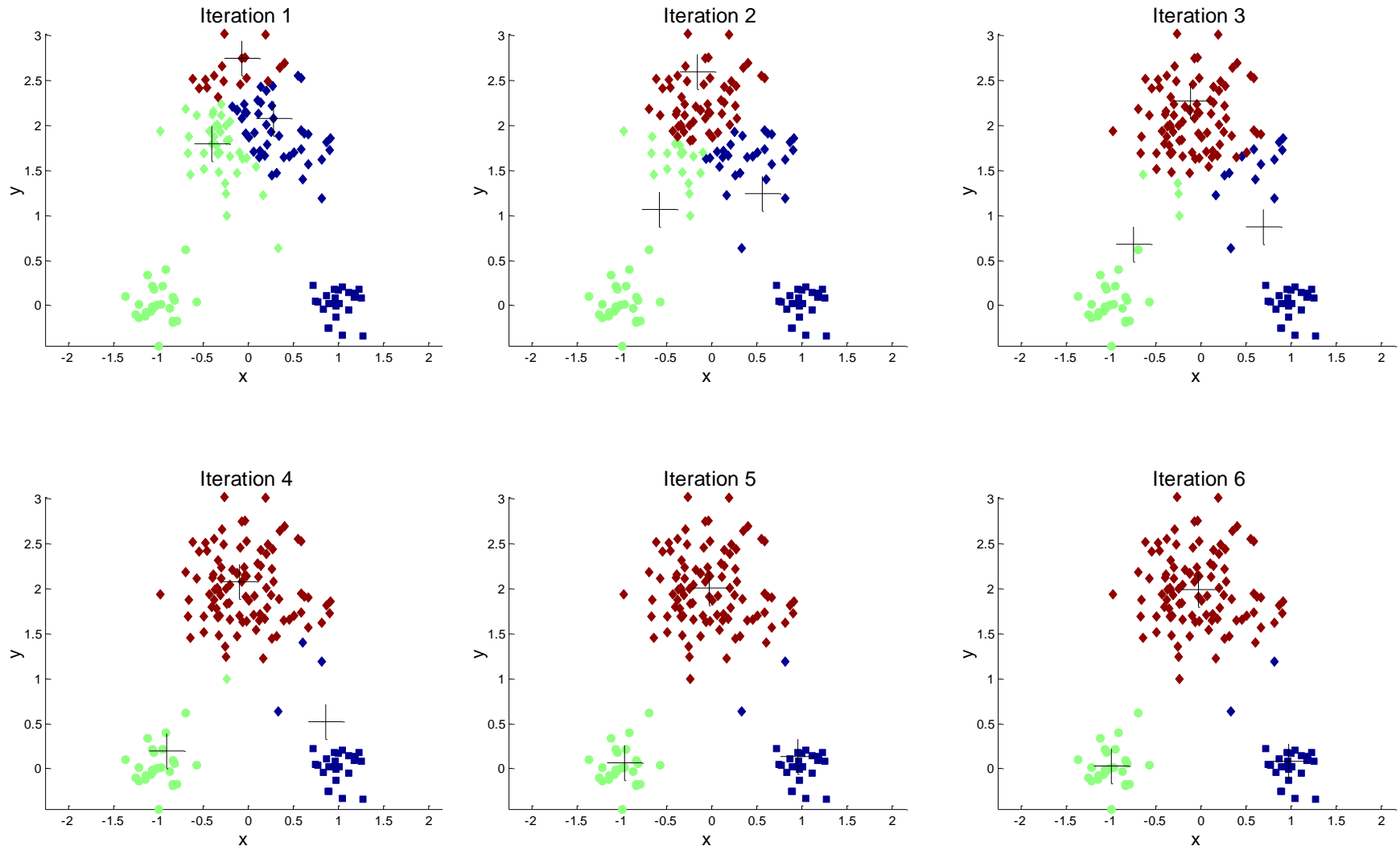


Sub-optimal Clustering

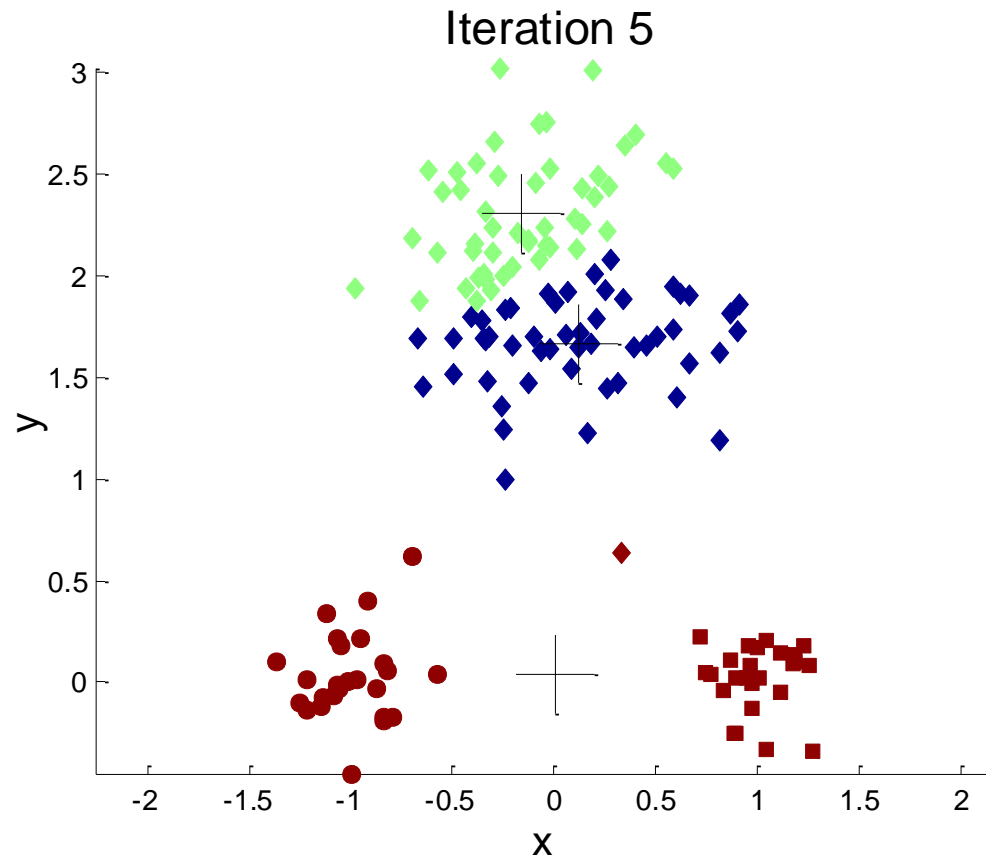
Importance of Choosing Initial Centroids



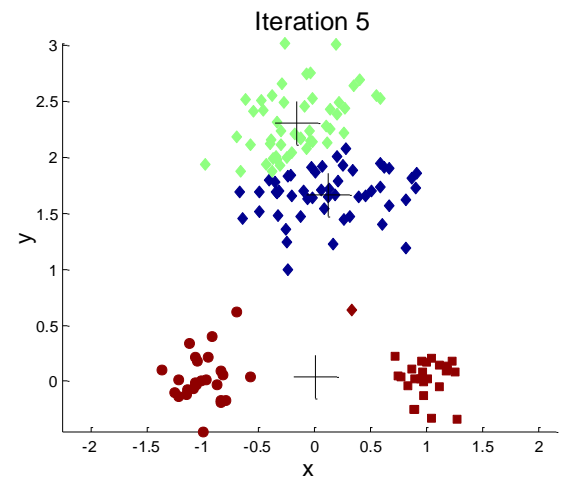
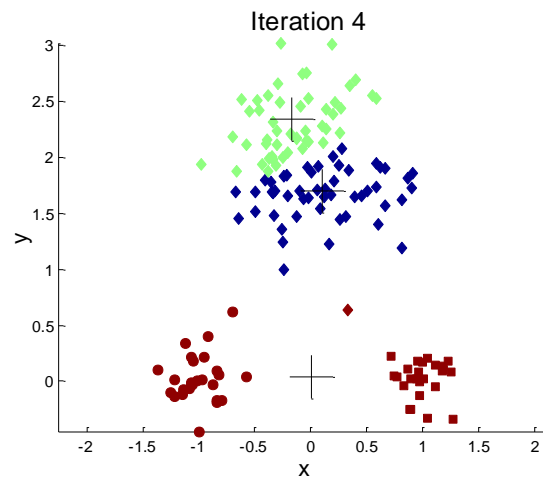
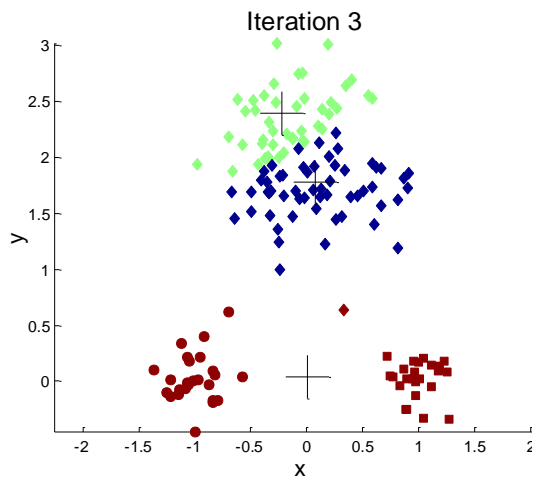
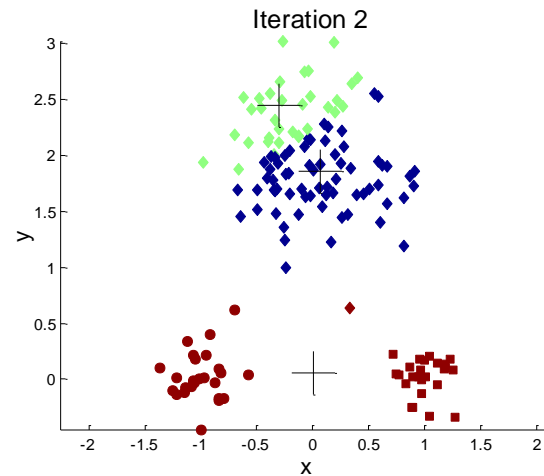
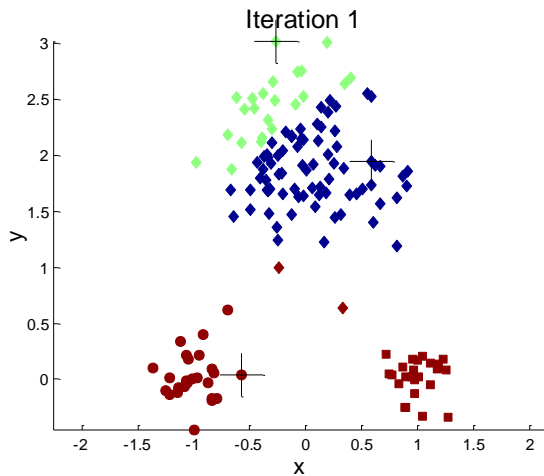
Importance of Choosing Initial Centroids



Importance of Choosing Initial Centroids



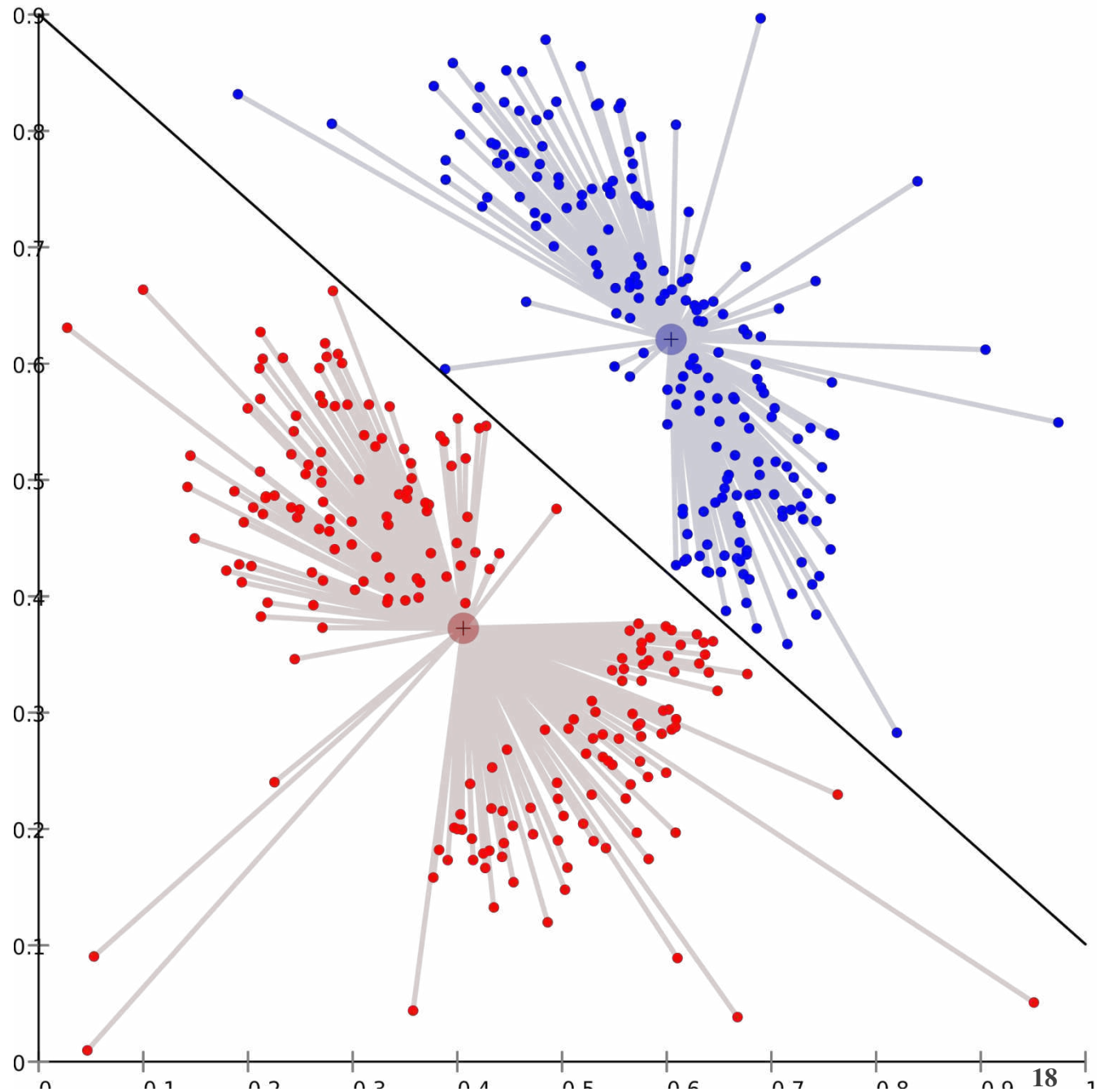
Importance of Choosing Initial Centroids



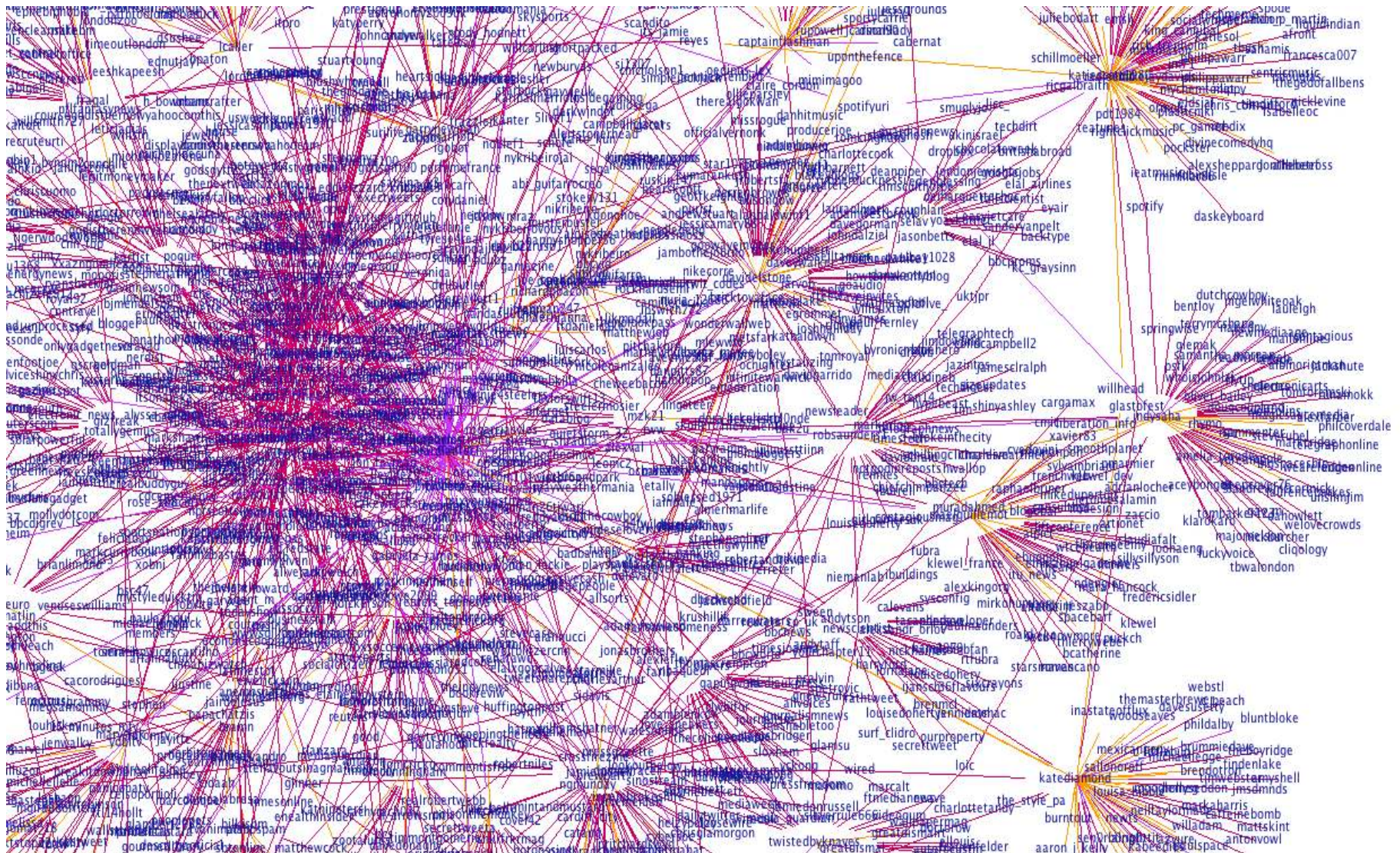
Solutions to Initial Centroids Problem

- 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:

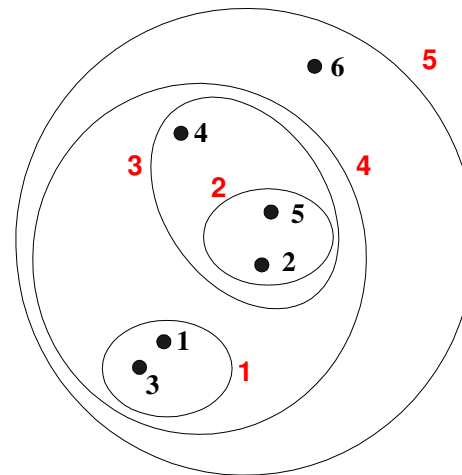
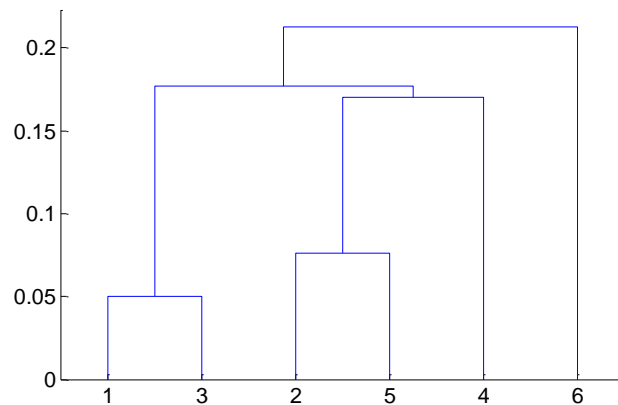


What do to in high-dimensional data?



Hierarchical Clustering

- Produces a set of nested clusters organized as a hierarchical tree
- Can be visualized as a dendrogram
 - A tree like diagram that records the sequences of merges or splits



Strengths of Hierarchical Clustering

- Do not have to assume any particular number of clusters
 - Any desired number of clusters can be obtained by 'cutting' the dendrogram at the proper level

Hierarchical Clustering

- Two main types of hierarchical clustering
 - Agglomerative:
 - Start with the points as individual clusters
 - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
 - Divisive:
 - Start with one, all-inclusive cluster
 - At each step, split a cluster until each cluster contains a point (or there are k clusters)
- Traditional hierarchical algorithms use a similarity or distance matrix
 - Merge or split one cluster at a time

Agglomerative Clustering Algorithm

More popular hierarchical clustering technique.

Basic algorithm is straightforward:

1. Compute the proximity matrix
2. Let each data point be a cluster
3. **Repeat**
4. Merge the two closest clusters
5. Update the proximity matrix
6. **Until** only a single cluster remains

Key operation is the computation of the proximity of two clusters

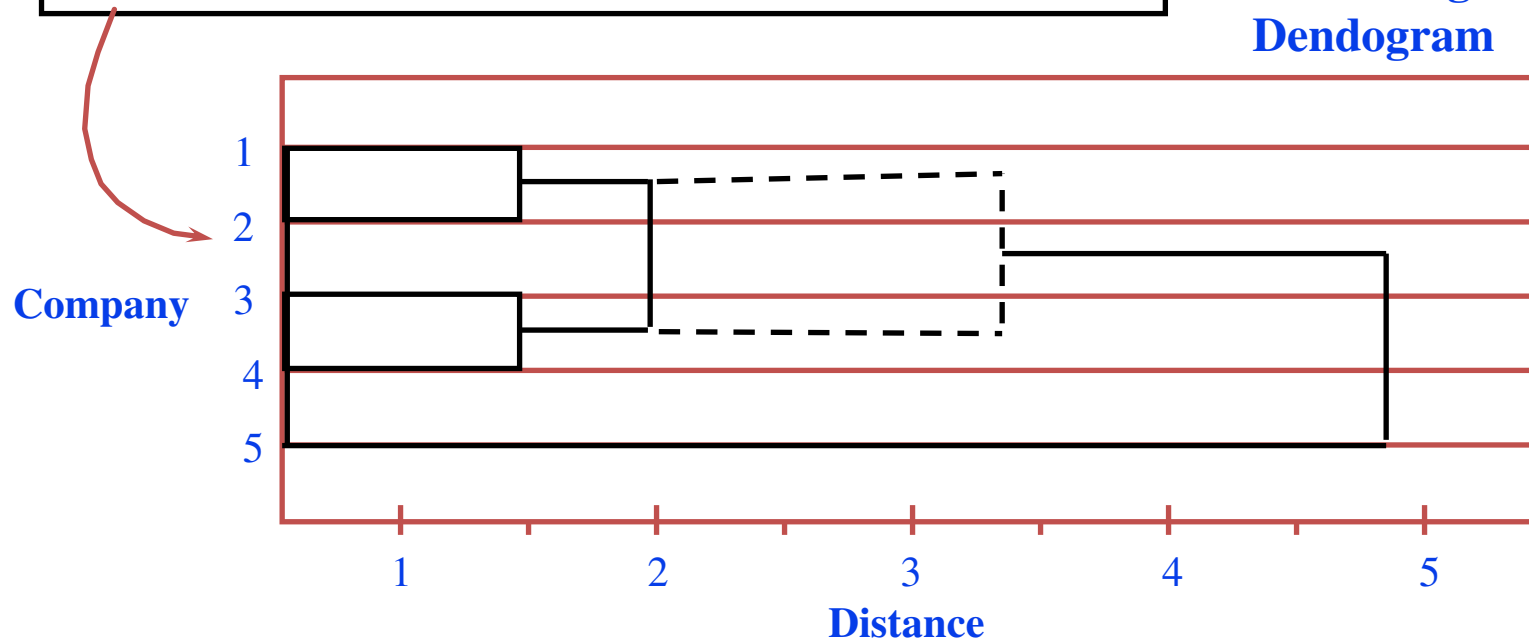
- Different approaches to defining the distance between clusters distinguish the different algorithms

Cluster Example

Distance Matrix

	Co#1	Co#2	Co#3	Co#4	Co#5
Company #1	0.00				
Company #2	1.49	0.00			
Company #3	3.42	2.29	0.00		
Company #4	1.81	1.99	1.48	0.00	
Company #5	5.05	4.82	4.94	4.83	0.00

Resulting
Dendrogram

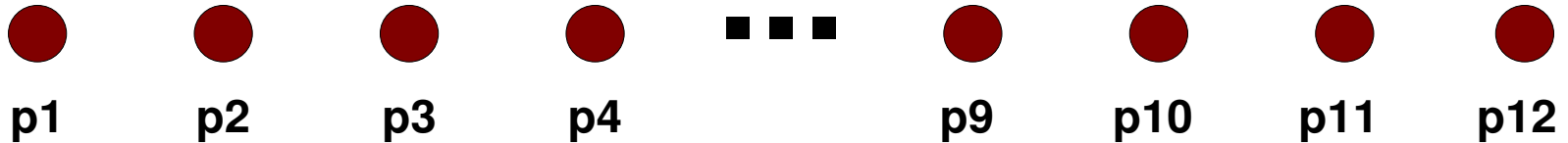


Starting Situation

- Start with clusters of individual points and a proximity matrix

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

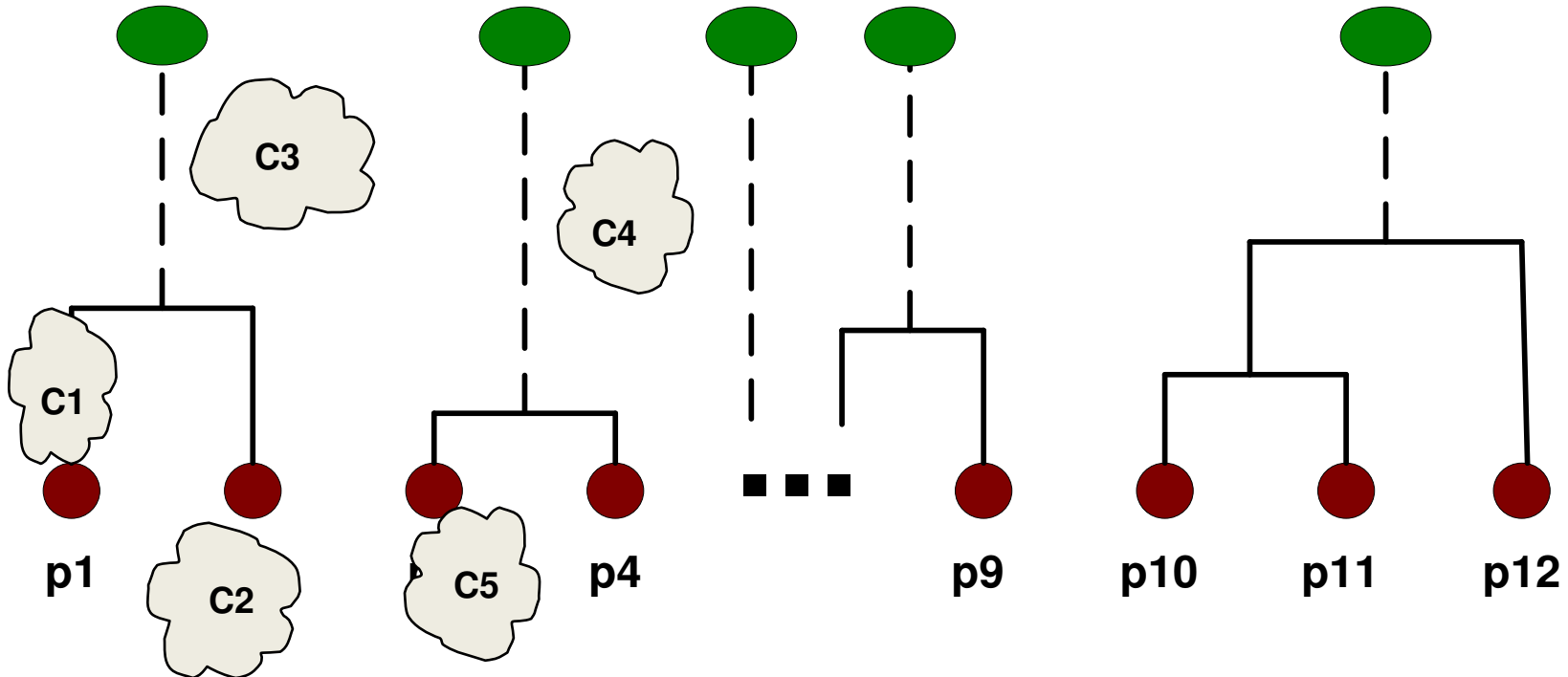


Intermediate Situation

After some merging steps, we have some clusters

	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix

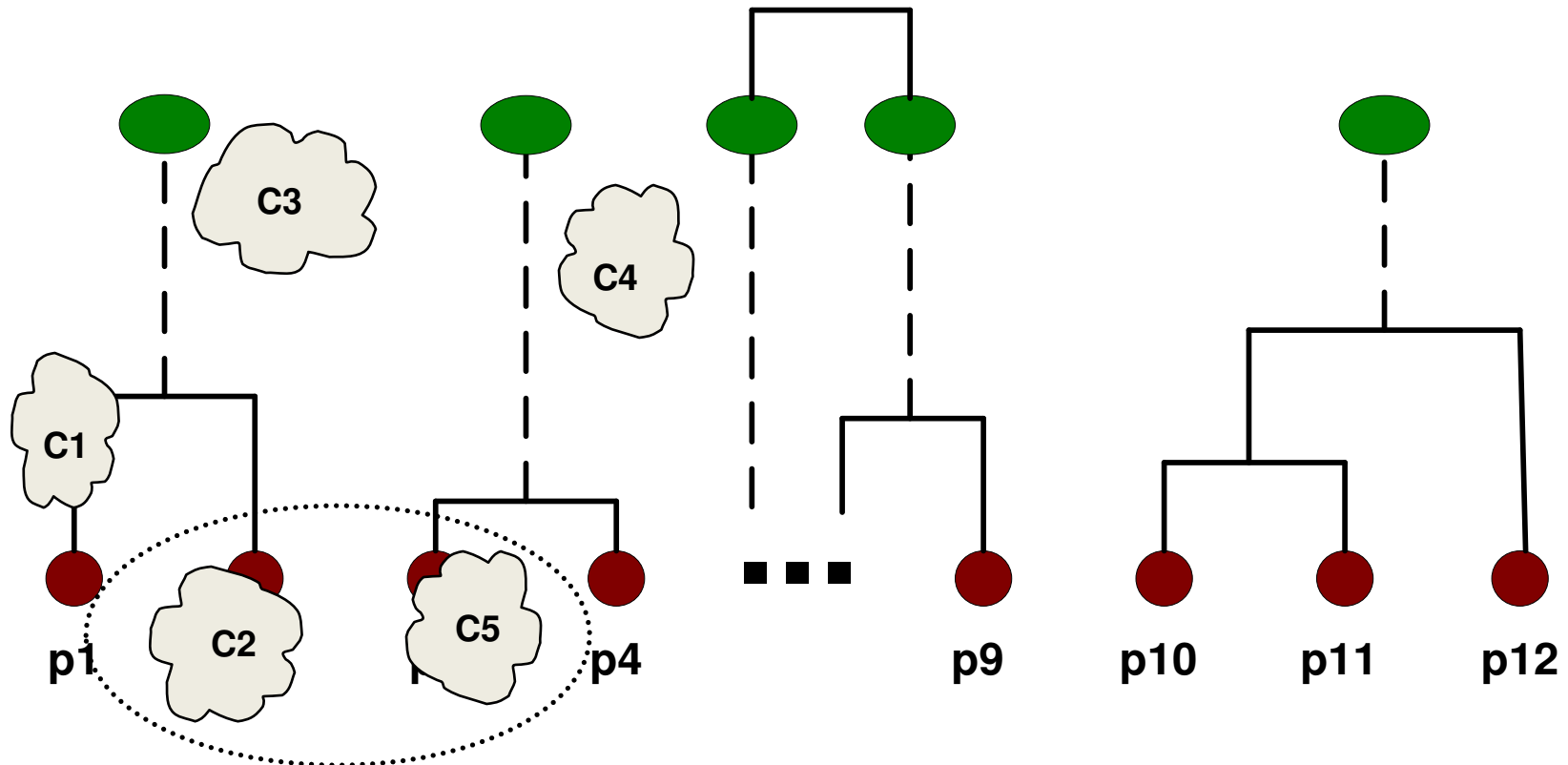


Intermediate Situation

We want to merge the two closest clusters (C2 and C5) and update the proximity matrix.

	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix

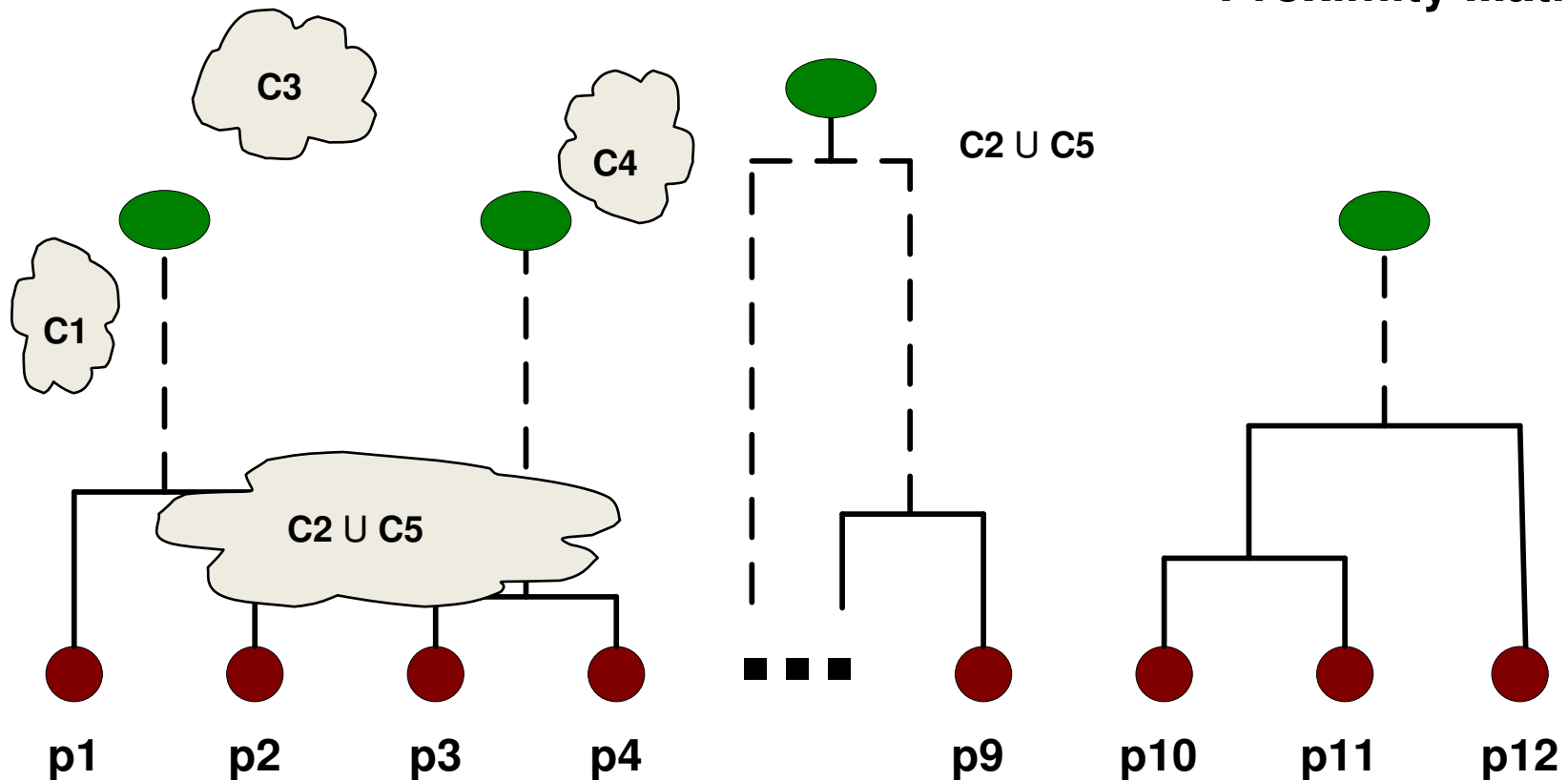


After Merging

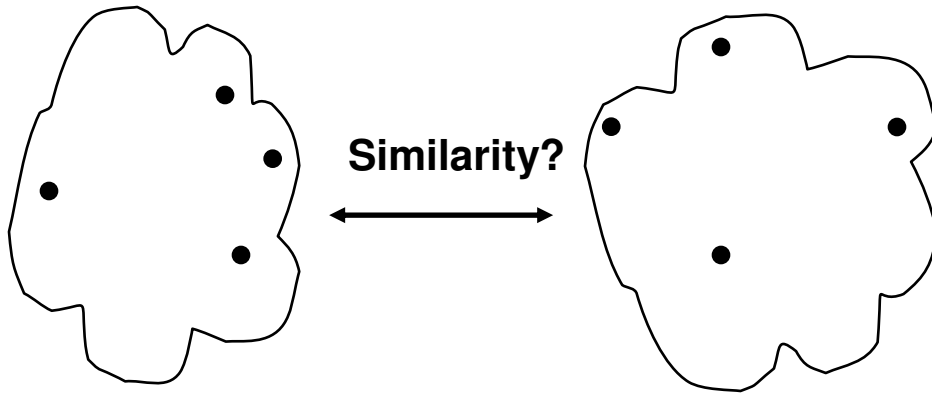
The question is “How do we update the proximity matrix?”

		C2 U C5	C3	C4
	C1	C5	C3	C4
C1		?		
	?	?	?	?
C3		?		
C4		?		

Proximity Matrix



How to Define Inter-Cluster Similarity

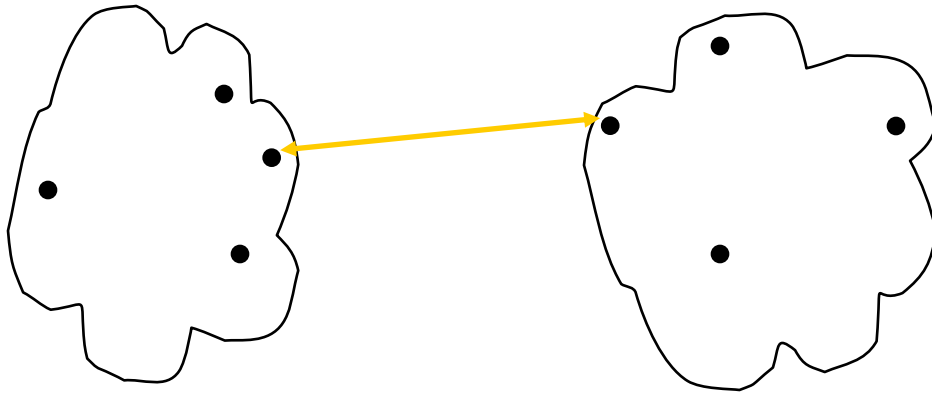


- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
 - Ward's Method uses squared error

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						

· **Proximity Matrix**

How to Define Inter-Cluster Similarity

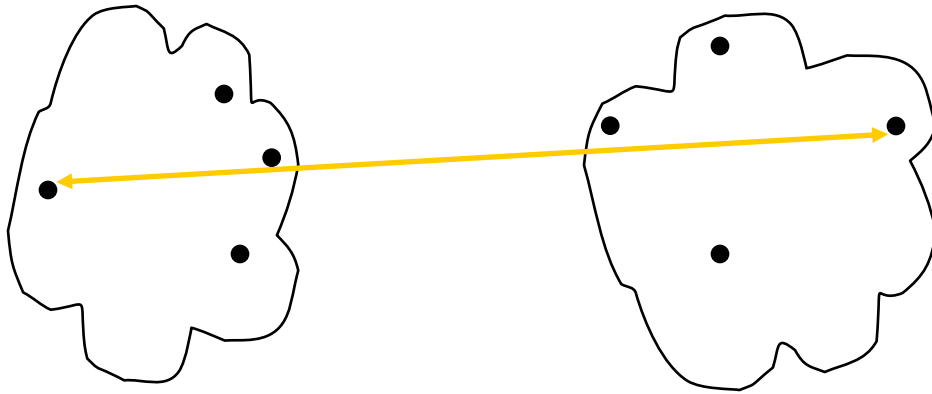


- ❑ MIN
- ❑ MAX
- ❑ Group Average
- ❑ Distance Between Centroids
- ❑ Other methods driven by an objective function
 - Ward's Method uses squared error

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

How to Define Inter-Cluster Similarity

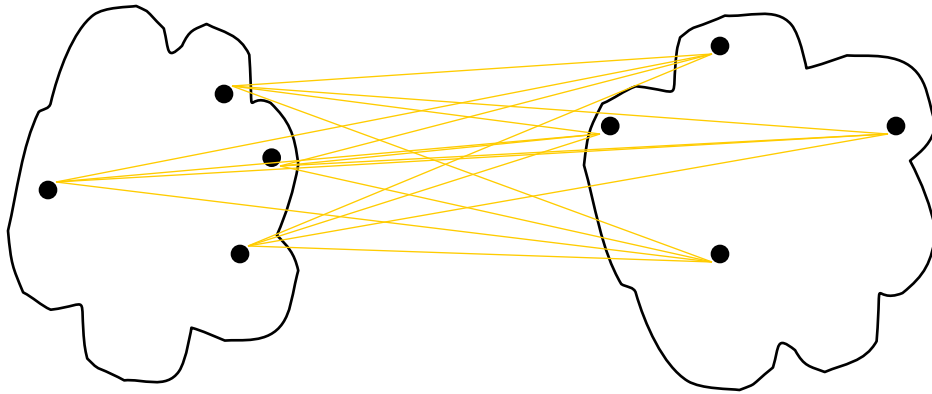


- ❑ MIN
- ❑ MAX
- ❑ Group Average
- ❑ Distance Between Centroids
- ❑ Other methods driven by an objective function
 - Ward's Method uses squared error

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

How to Define Inter-Cluster Similarity

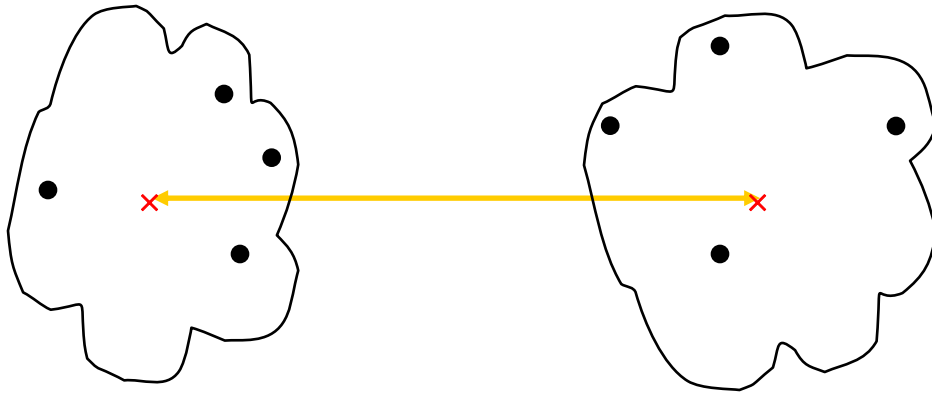


- ❑ MIN
- ❑ MAX
- ❑ **Group Average**
- ❑ Distance Between Centroids
- ❑ Other methods driven by an objective function
 - Ward's Method uses squared error

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

How to Define Inter-Cluster Similarity



- ❑ MIN
- ❑ MAX
- ❑ Group Average
- ❑ Distance Between Centroids
- ❑ Other methods driven by an objective function
 - Ward's Method uses squared error

	p1	p2	p3	p4	p5	...
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

Cluster Similarity: Ward's Method

- Similarity of two clusters is based on the increase in squared error when two clusters are merged
 - Similar to group average if distance between points is distance squared
- Less susceptible to noise and outliers
- Biased towards globular clusters
- Hierarchical analogue of K-means
 - Can be used to initialize K-means

Hierarchical Clustering: Problems and Limitations

- Computational complexity in time and space
- Once a decision is made to combine two clusters, it cannot be undone
- No objective function is directly minimized
- Different schemes have problems with one or more of the following:
 - Sensitivity to noise and outliers
 - Difficulty handling different sized clusters and convex shapes
 - Breaking large clusters

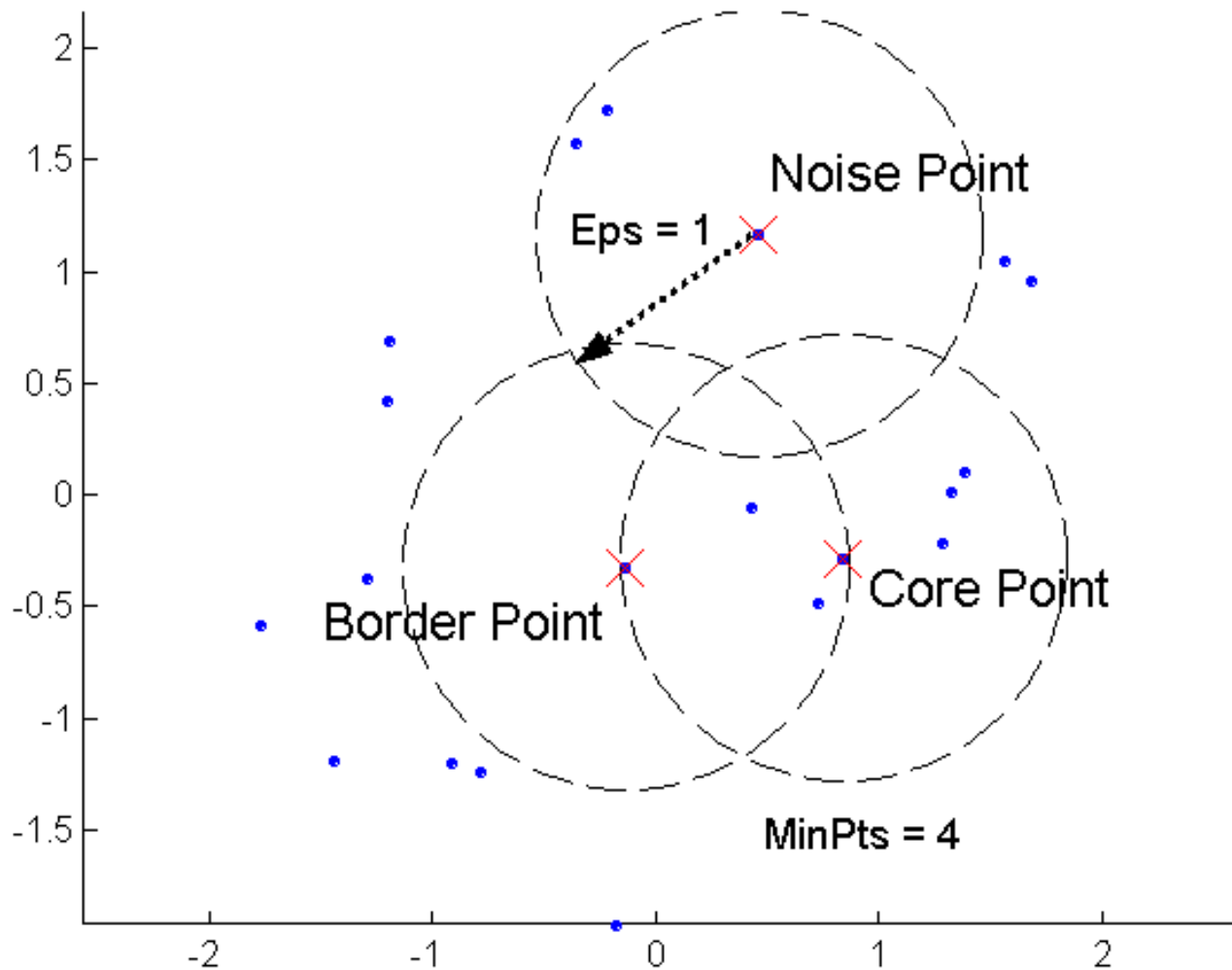
DBSCAN: Density-Based Clustering

- DBSCAN is a Density-Based Clustering algorithm
- Reminder: In density based clustering we partition points into dense regions separated by not-so-dense regions.
- Important Questions:
 - How do we measure density?
 - What is a dense region?
- DBSCAN:
 - Density at point p : number of points within a circle of radius Eps
 - Dense Region: A circle of radius Eps that contains at least $MinPts$ points

DBSCAN

- Characterization of points
 - A point is a **core point** if it has more than a specified number of points (**MinPts**) within **Eps**
 - These points belong in a **dense region** and are at the **interior** of a cluster
 - A **border point** has fewer than **MinPts** within **Eps**, but is in the neighborhood of a **core** point.
 - A **noise point** is any point that is not a core point or a border point.

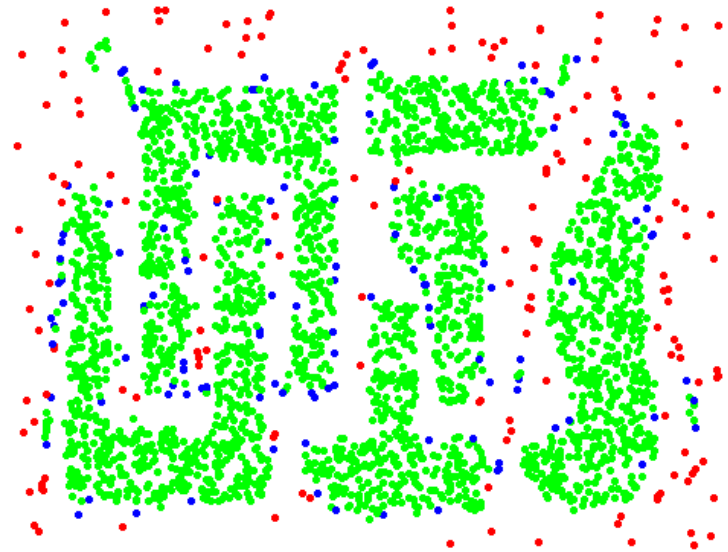
DBSCAN: Core, Border, and Noise Points



DBSCAN: Core, Border and Noise Points



Original Points



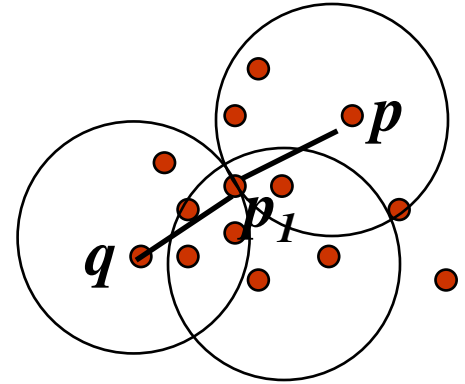
Point types: core,
border and noise

Eps = 10, MinPts = 4

Density-Connected points

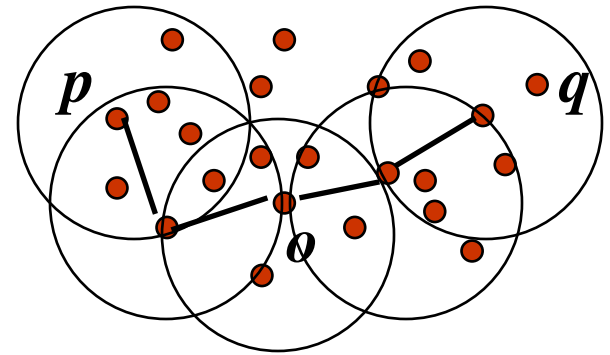
- Density edge

- We place an **edge** between two core points **q** and **p** if they are within distance **Eps**.



- Density-connected

- A point **p** is **density-connected** to a point **q** if there is a **path of edges** from **p** to **q**

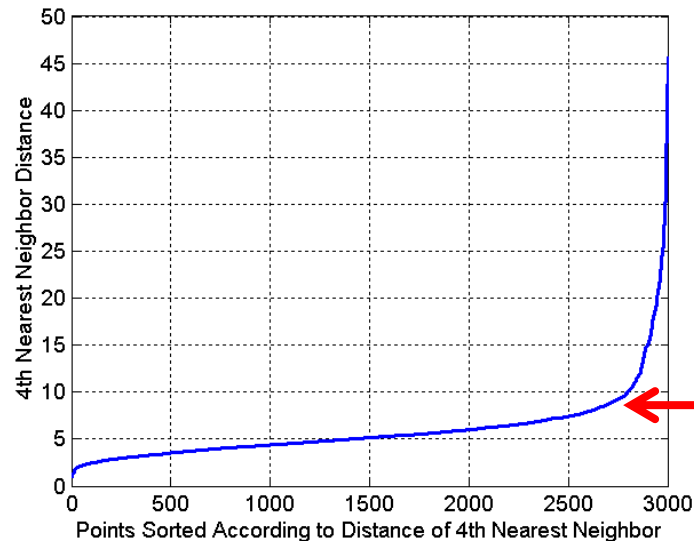


DBSCAN Algorithm

- Label points as **core**, **border** and **noise**
- Eliminate **noise** points
- For every **core** point **p** that has not been assigned to a cluster
 - Create a new cluster with the point **p** and all the points that are **density-connected** to **p**.
- Assign **border** points to the cluster of the closest core point.

DBSCAN: Determining Eps and MinPts

- Idea is that for points in a cluster, their k^{th} nearest neighbors are at roughly the same distance
- Noise points have the k^{th} nearest neighbor at farther distance
- So, plot sorted distance of every point to its k^{th} nearest neighbor
- Find the distance d where there is a “knee” in the curve
 - $\text{Eps} = d$, $\text{MinPts} = k$

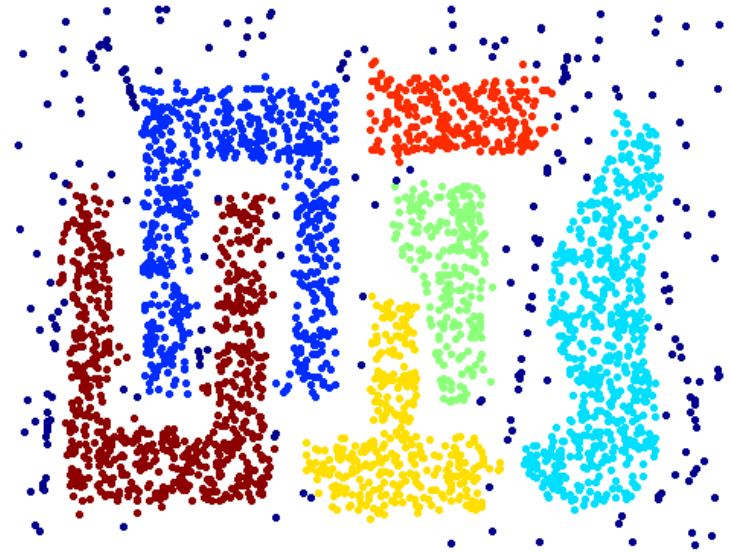


Eps ~ 7-10
MinPts = 4

When DBSCAN Works Well



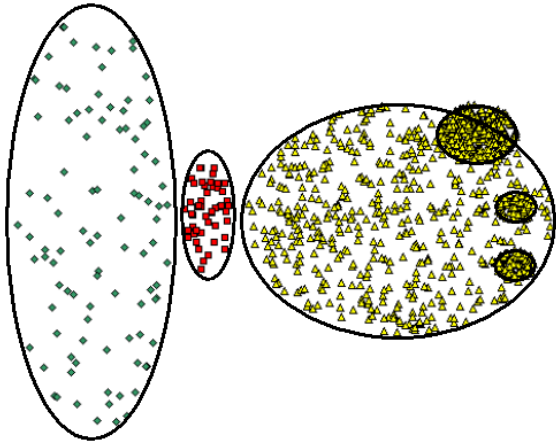
Original Points



Clusters

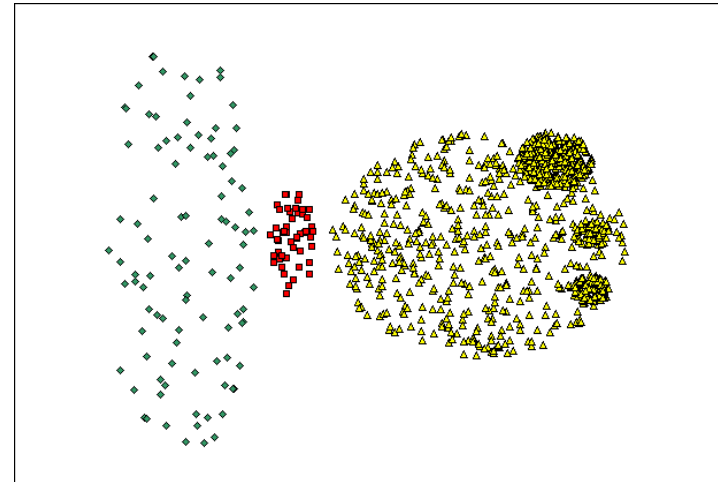
- Resistant to Noise
- Can handle clusters of different shapes and sizes

When DBSCAN Does NOT Work Well

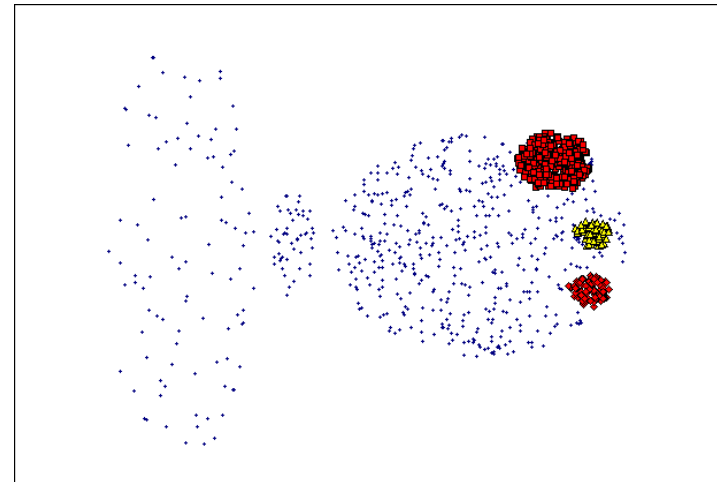


Original Points

- **Sensitive to Parameters**
- **Varying densities**
- **High-dimensional data**



(MinPts=4, Eps=9.75).



(MinPts=4, Eps=9.92)