Lecture 4: Clustering

Machine Learning Paradigm

- Observe set of examples: training data
- Infer something about process that generated that data
- Use inference to make predictions about previously unseen data: test data
- Supervised: given a set of feature/label pairs, find a rule that predicts the label associated with a previously unseen input
- *Unsupervised*: given a set of feature vectors (without labels) group them into "natural clusters"

Clustering Is an Optimization Problem

$$variability(c) = \sum_{e \in c} distance(mean(c), e)^2$$

$$dissimilarity(C) = \sum_{c \in C} variability(c)$$

- Why not divide variability by size of cluster?
  - Big and bad worse than small and bad
- Is optimization problem finding a C that minimizes *dissimilarity(C)*?
  - No, otherwise could put each example in its own cluster
- Need a constraint, e.g.,
  - Minimum distance between clusters
  - Number of clusters

Two Popular Methods

- Hierarchical clustering
- K-means clustering

Types of Clusterings

A clustering is a set of clusters

Important distinction between hierarchical and partitional sets of clusters

Partitional Clustering

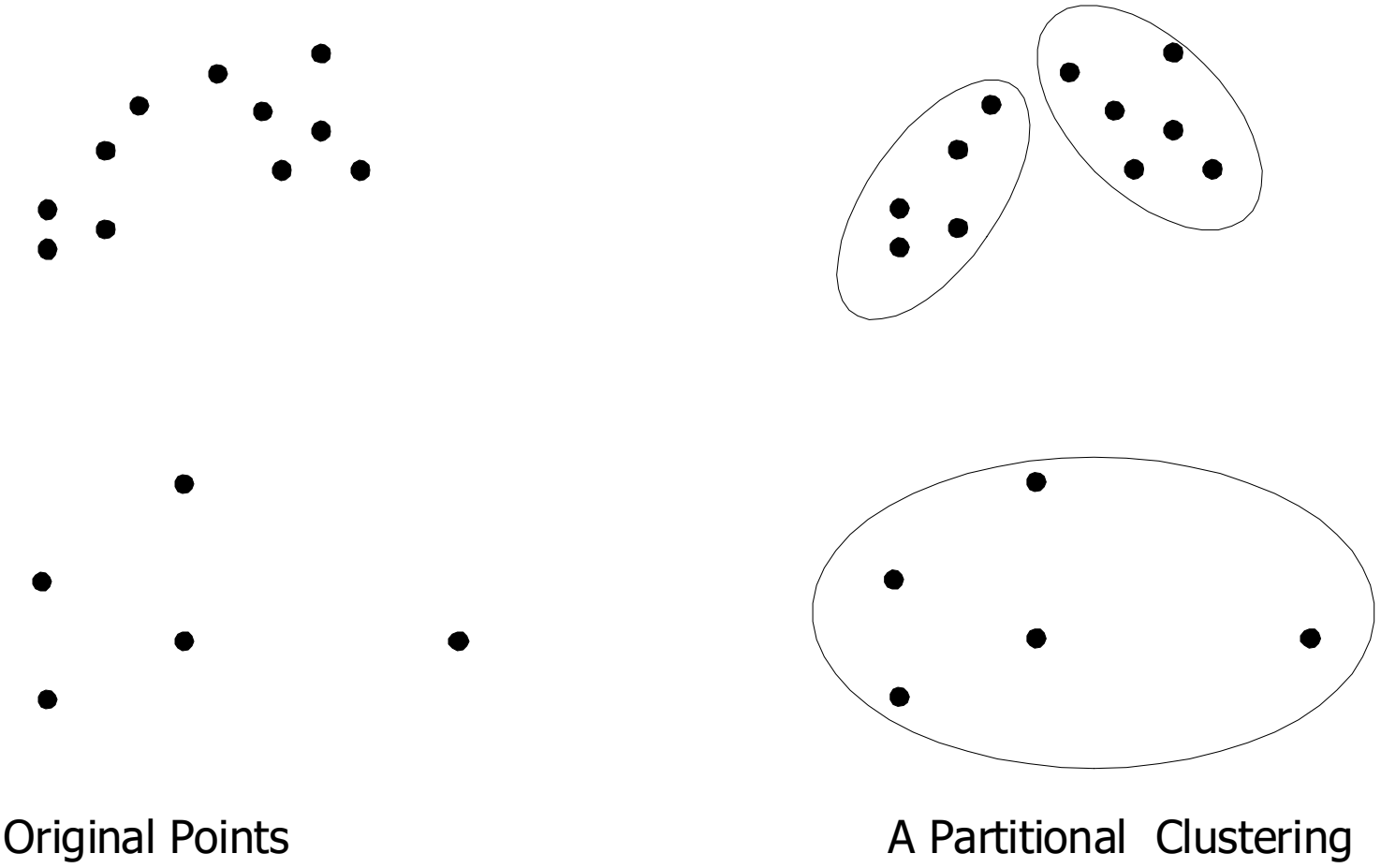
- Divides data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset

Hierarchical clustering

- A set of nested clusters organized as a hierarchical tree

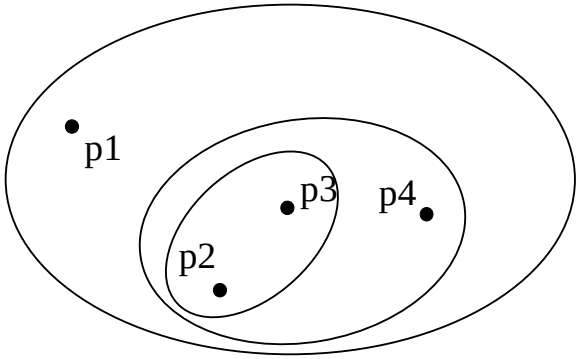
From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Partitional Clustering

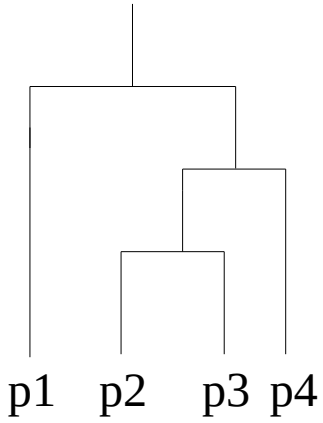


From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

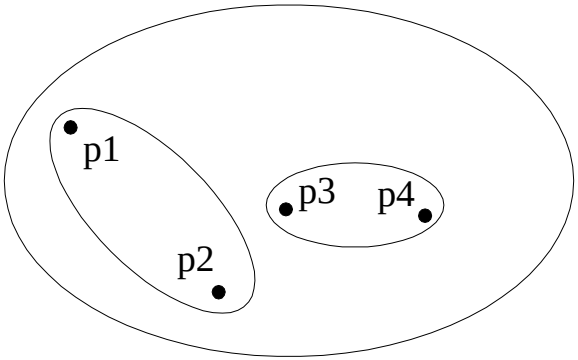
Hierarchical Clustering



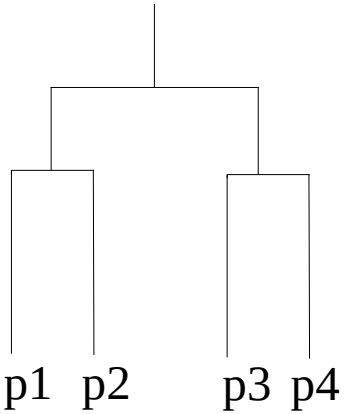
Traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Hierarchical Clustering



Non-traditional Dendrogram

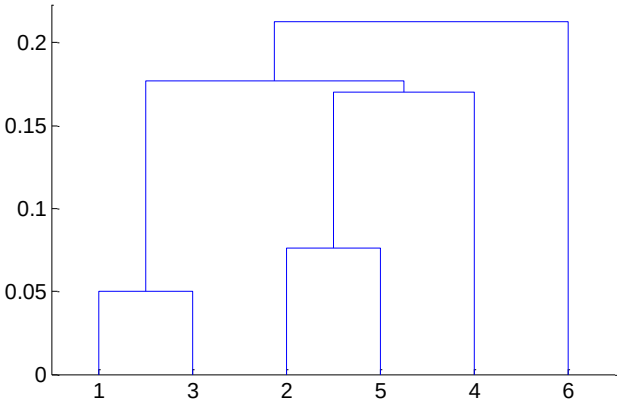
From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

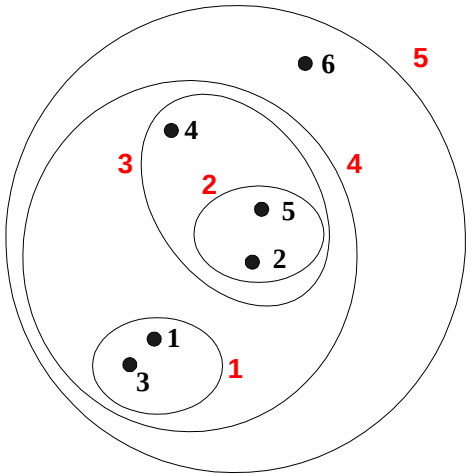
Hierarchical Clustering

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





From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

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 dendogram at the proper level

They may correspond to meaningful taxonomies

- Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, …)

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Hiearchical Clustering

1. Start by assigning each item to a cluster, so that if you have N items, you now have N clusters, each containing just one item.

2. Find the closest (most similar) pair of clusters and merge them into a single cluster, so that now you have one fewer cluster.

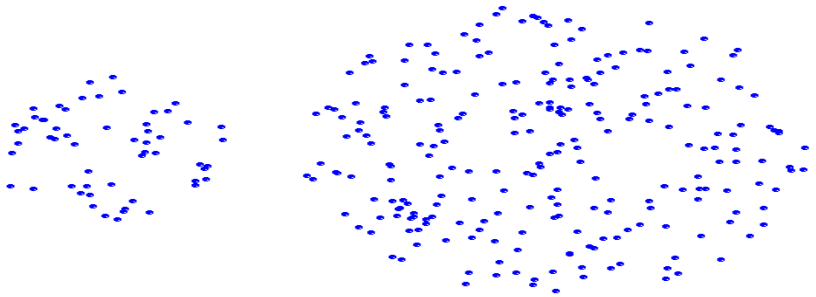
3. Continue the process until all items are clustered into a single cluster of size N.

What does distance mean?

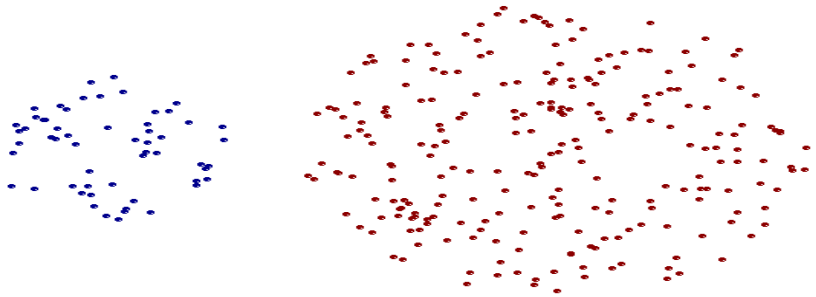
Linkage Metrics

- S*ingle-linkage:* consider the distance between one cluster and another cluster to be equal to the shortest (MIN) distance from any member of one cluster to any member of the other cluster
- C*omplete-linkage*: consider the distance between one cluster and another cluster to be equal to the greatest (MAX) distance from any member of one cluster to any member of the other cluster
- A*verage-linkage:* consider the distance between one cluster and another cluster to be equal to the average distance from any member of one cluster to any member of the other cluster

Strength of MIN



Original Points

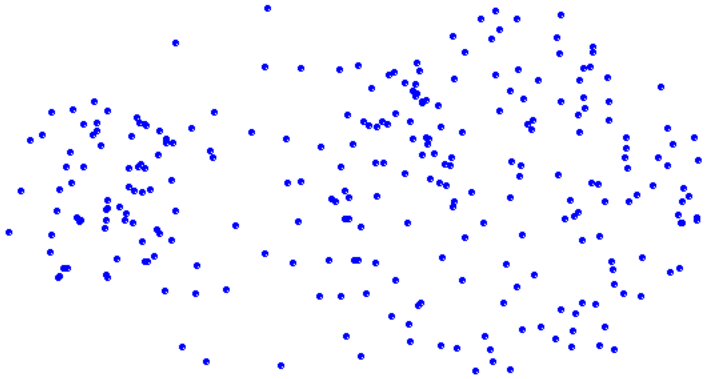


Two Clusters

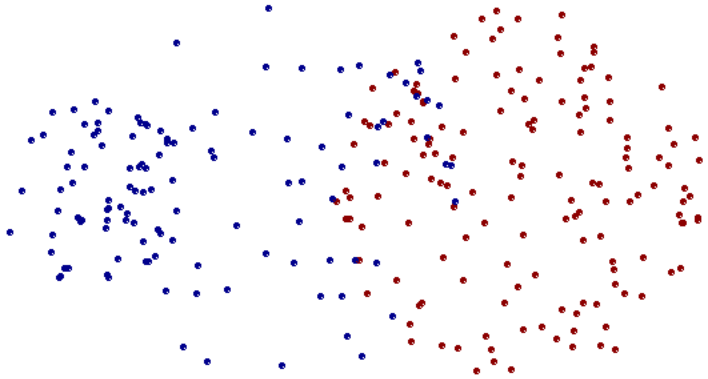
- Can handle non-elliptical shapes

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Limitations of MIN



Original Points

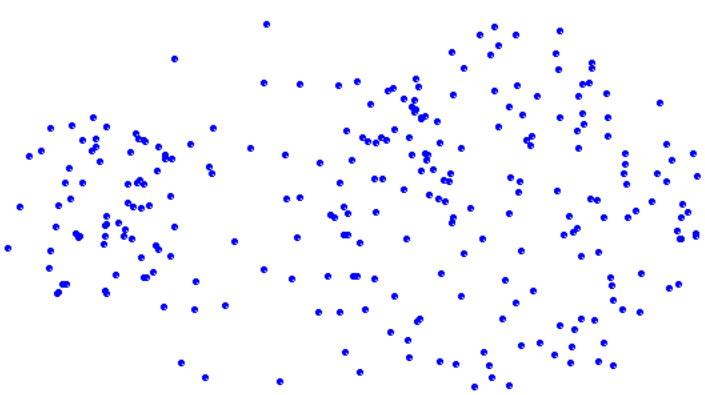


Two Clusters

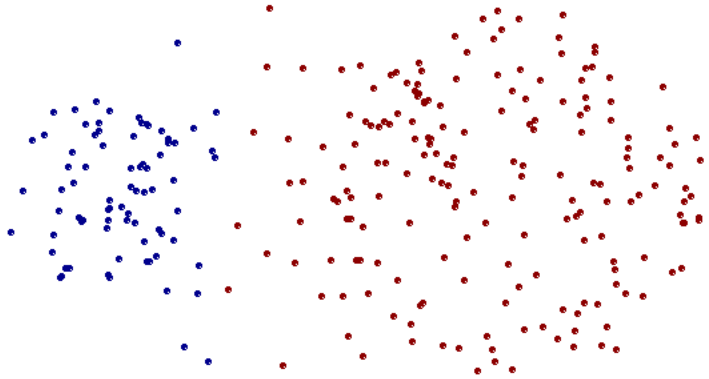
- Sensitive to noise and outliers

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Strength of MAX



Original Points

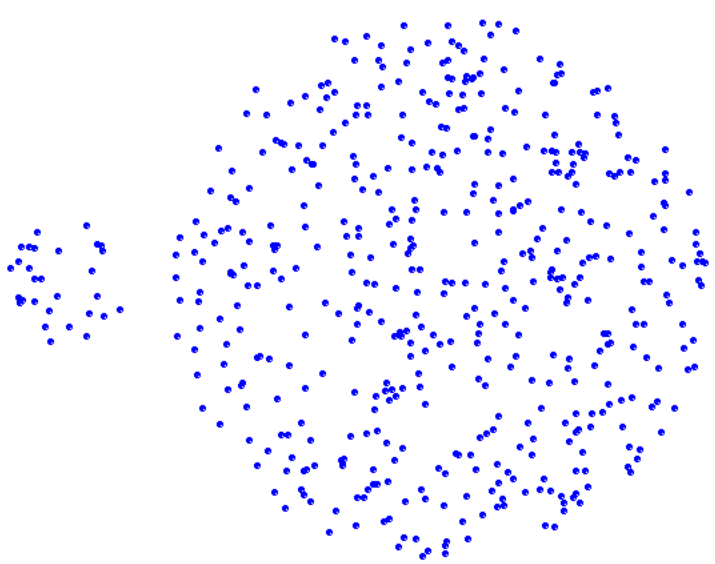


Two Clusters

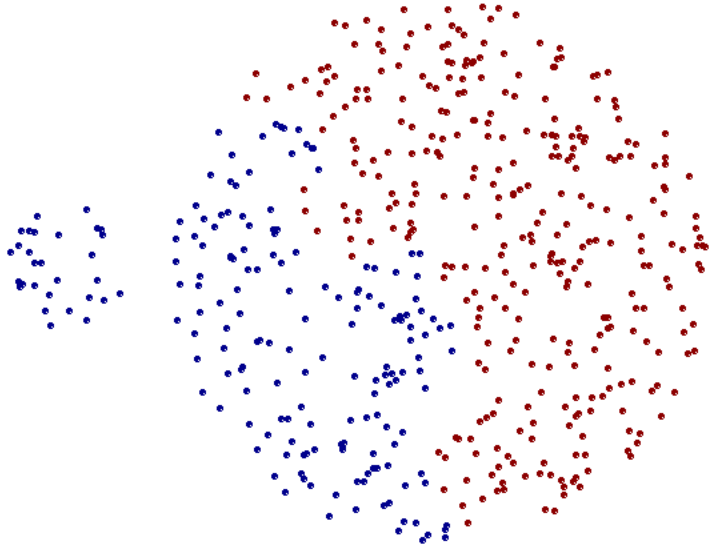
- Less susceptible to noise and outliers

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Limitations of MAX



Original Points



Two Clusters

- Tends to break large clusters
- Biased towards globular clusters

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Hierarchical Clustering: Group Average

Compromise between Single and Complete Link

Strengths

- Less susceptible to noise and outliers

Limitations

- Biased towards globular clusters

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Clustering Algorithms

- Hierarchical clustering
  - Can select number of clusters using dendogram
  - Deterministic
  - Flexible with respect to linkage criteria
  - Slow
    - Naïve algorithm n^3
    - n^2 algorithms exist for some linkage criteria
- K-means a much faster greedy algorithm
  - Most useful when you know how many clusters you want

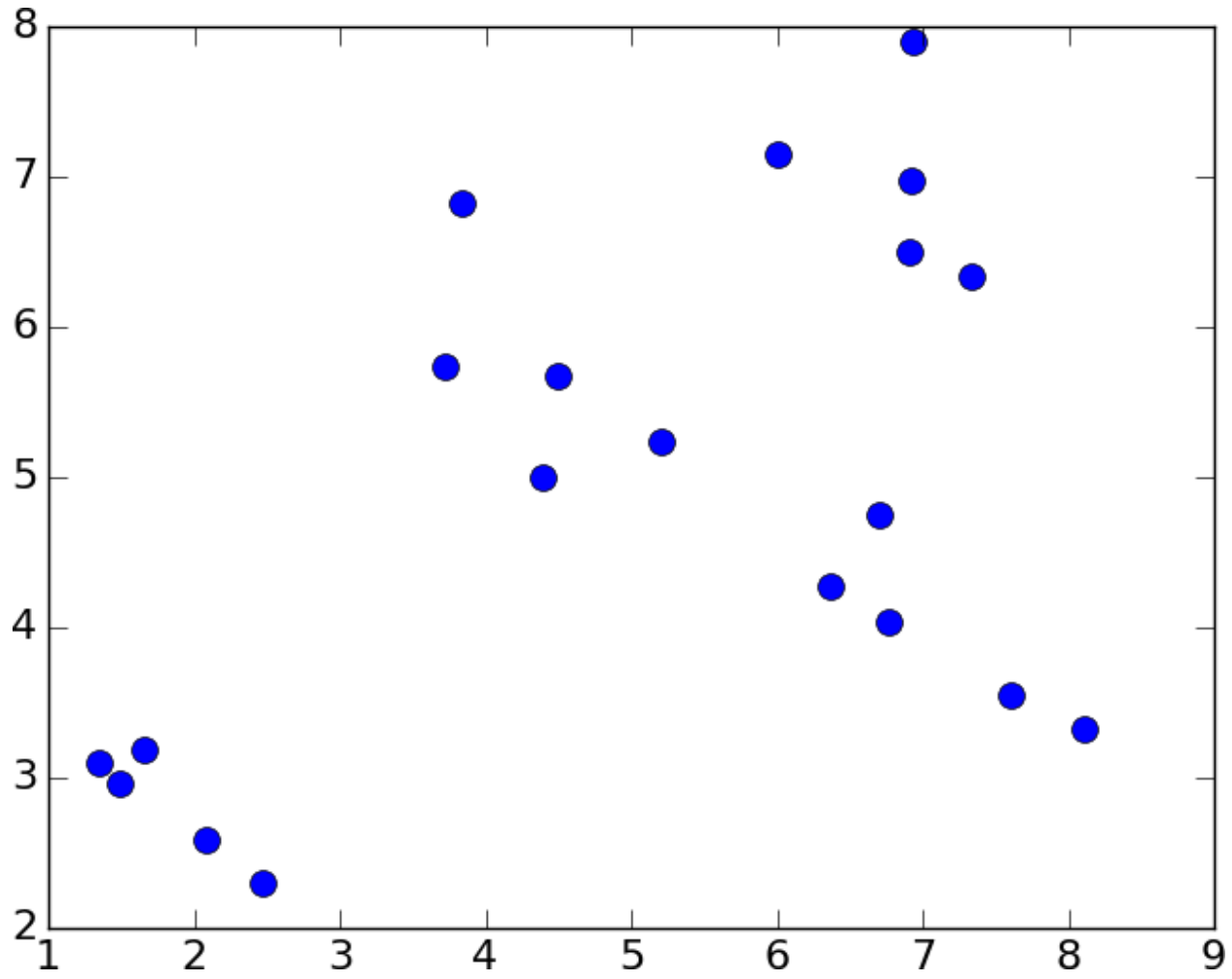
K-means Algorithm

```
randomly chose k examples as initial centroids
while true:
    create k clusters by assigning each
        example to closest centroid
    compute k new centroids by averaging
        examples in each cluster
    if centroids don't change:
        break
```

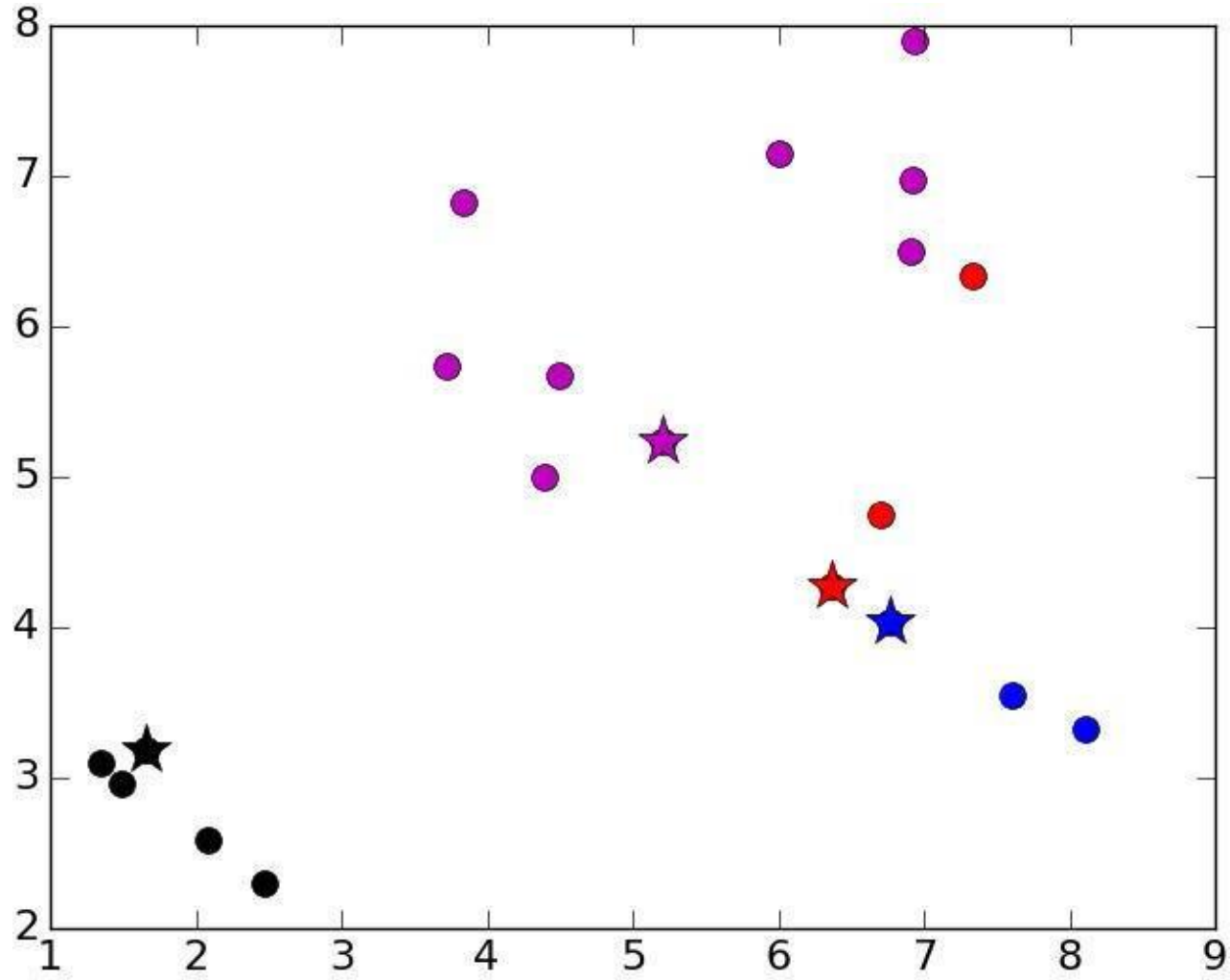
What is complexity of one iteration?

k*n*d, where n is number of points and d time required to compute the distance between a pair of points

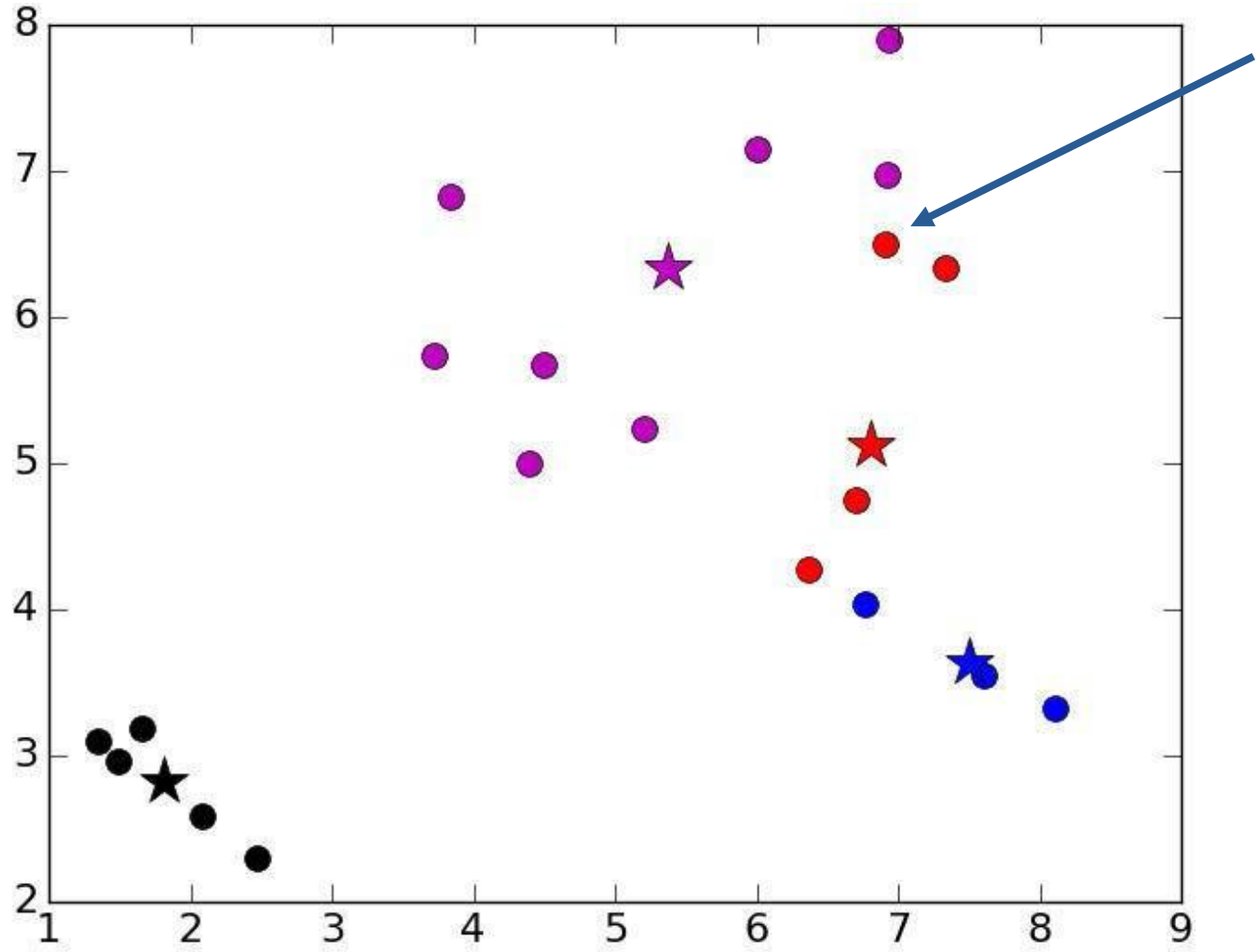
An Example



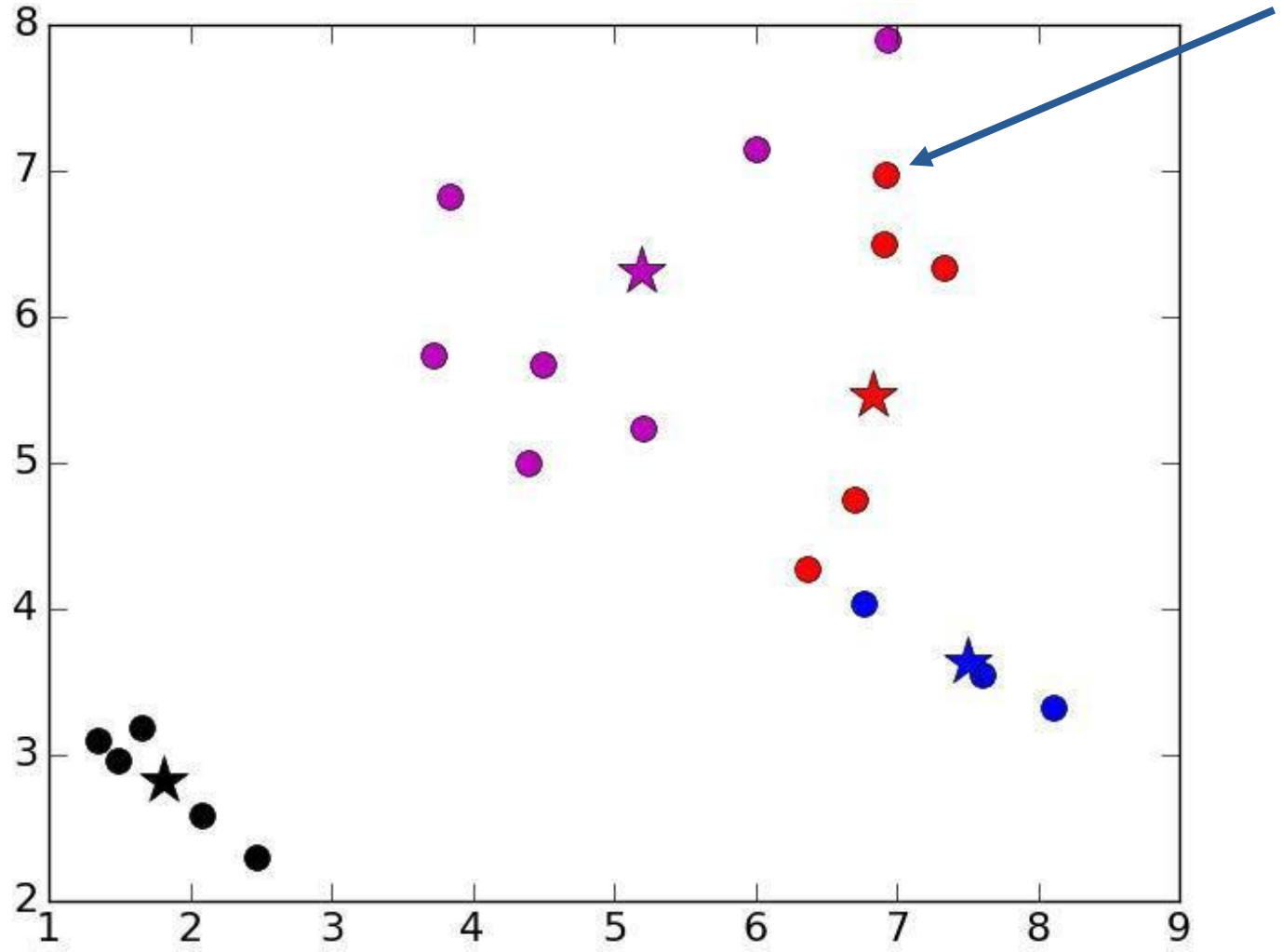
K = 4, Initial Centroids



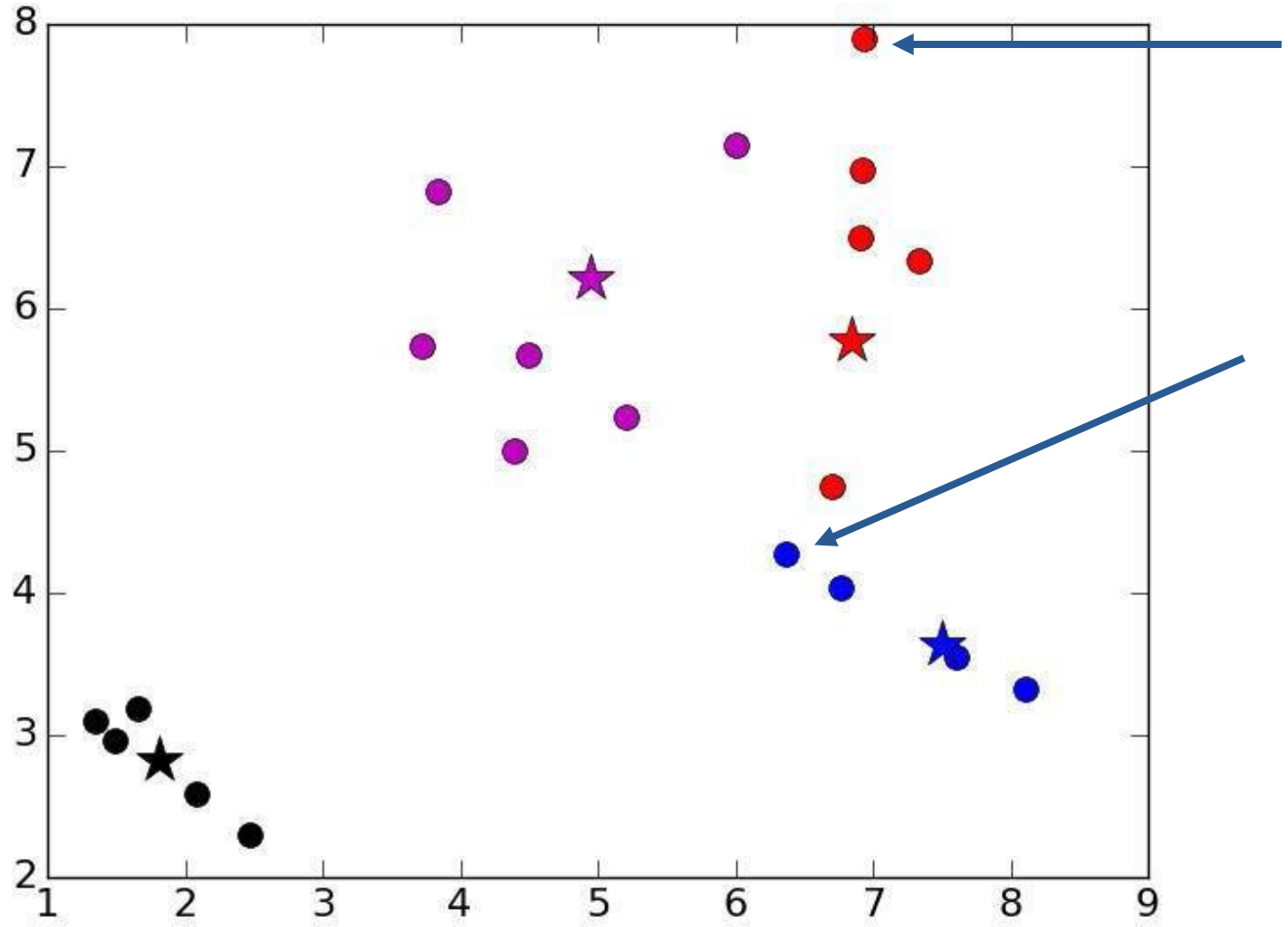
Iteration 1



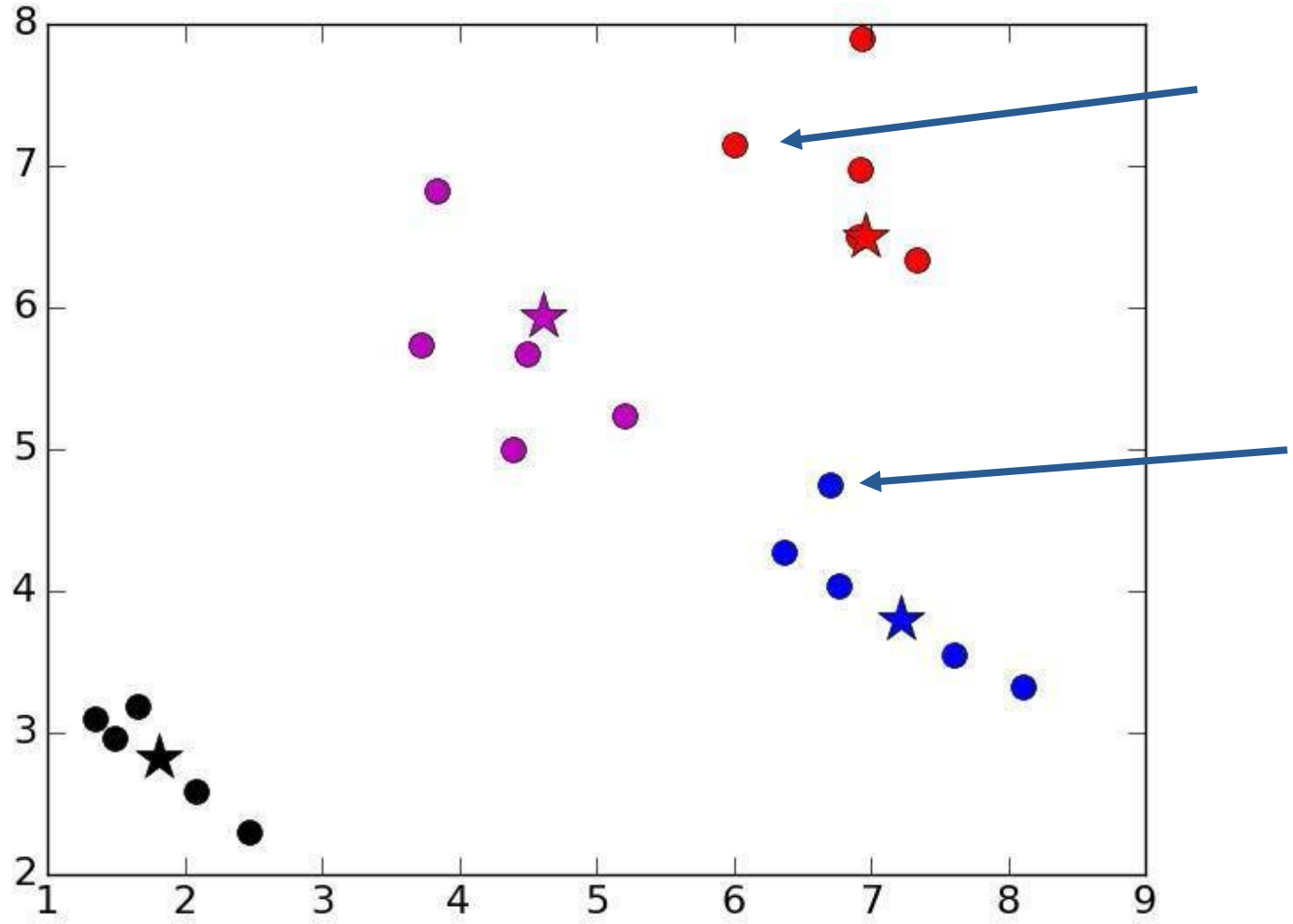
Iteration 2



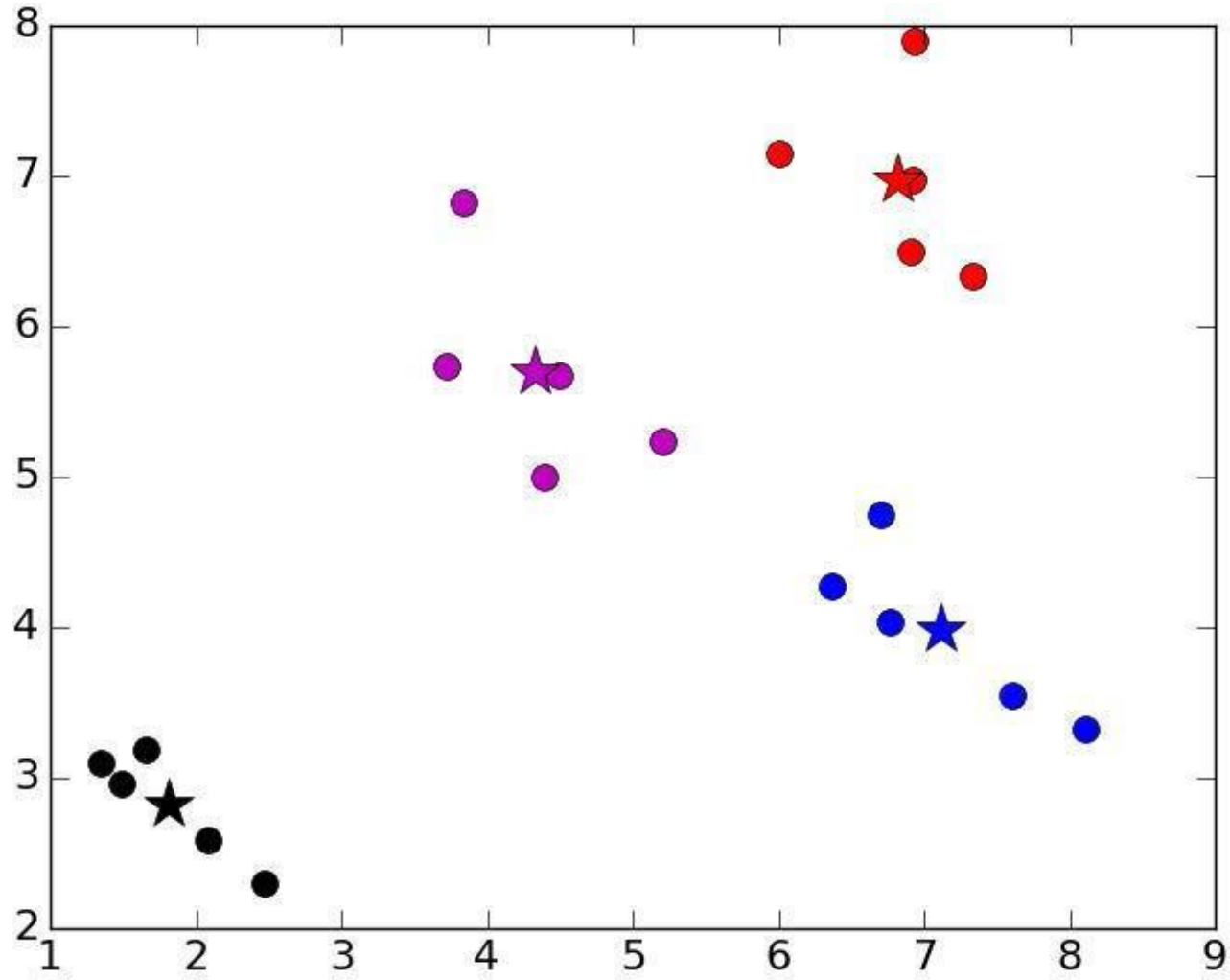
Iteration 3



Iteration 4

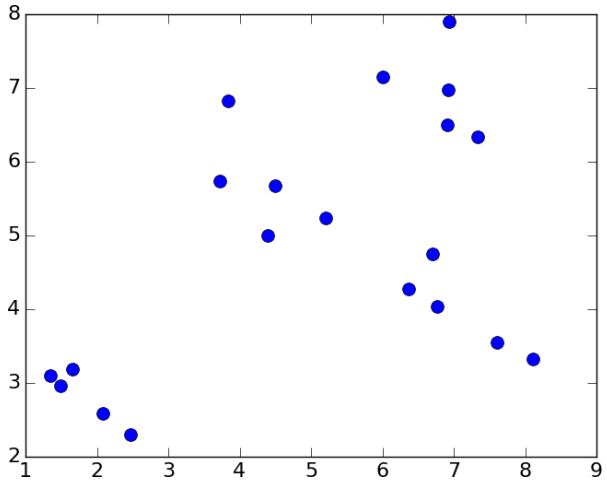


Iteration 5



Issues with k-means

- Choosing the “wrong” k can lead to strange results
  - Consider k = 3

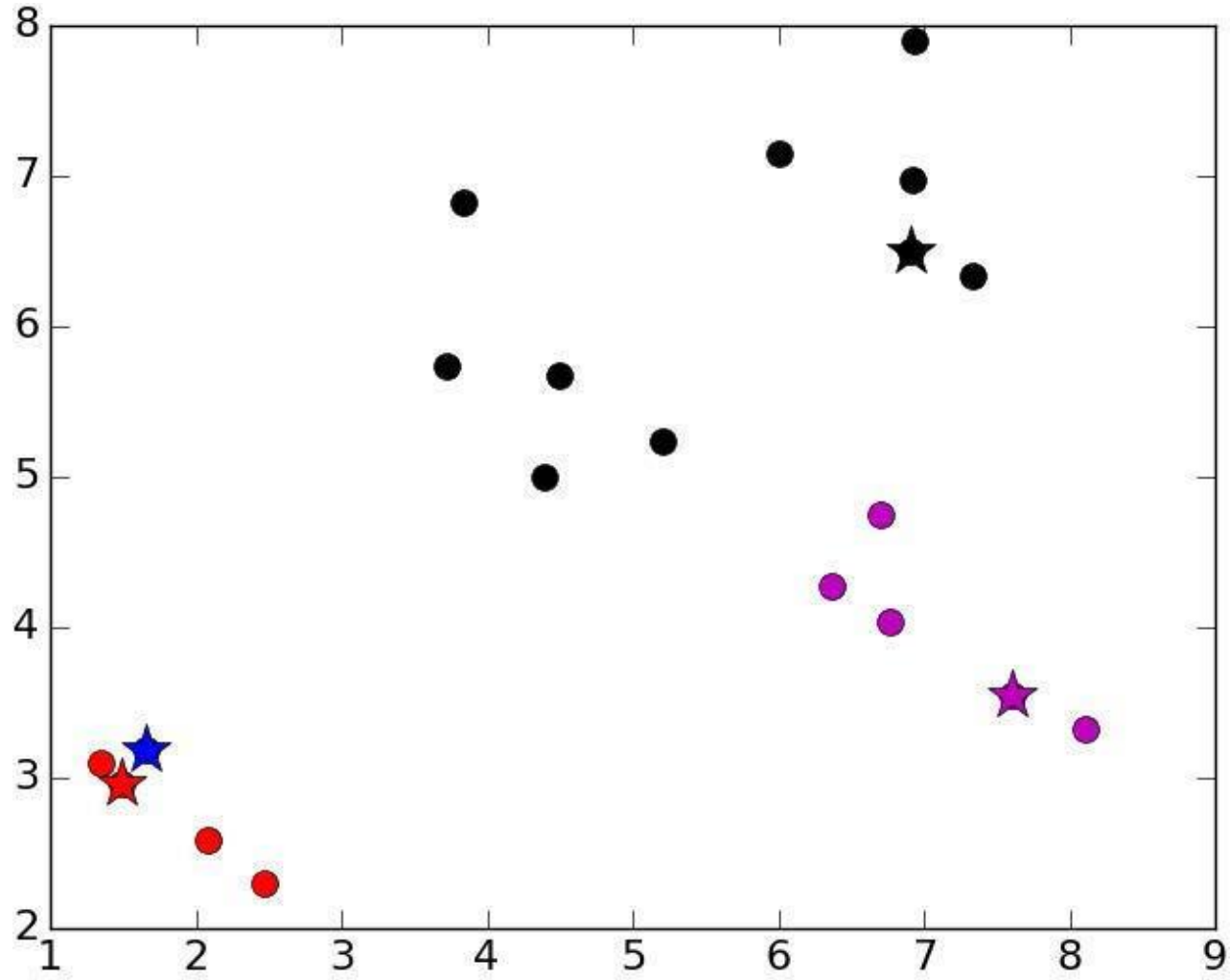


- Result can depend upon initial centroids
  - Number of iterations
  - Even final result
    - Greedy algorithm can find different local optimas

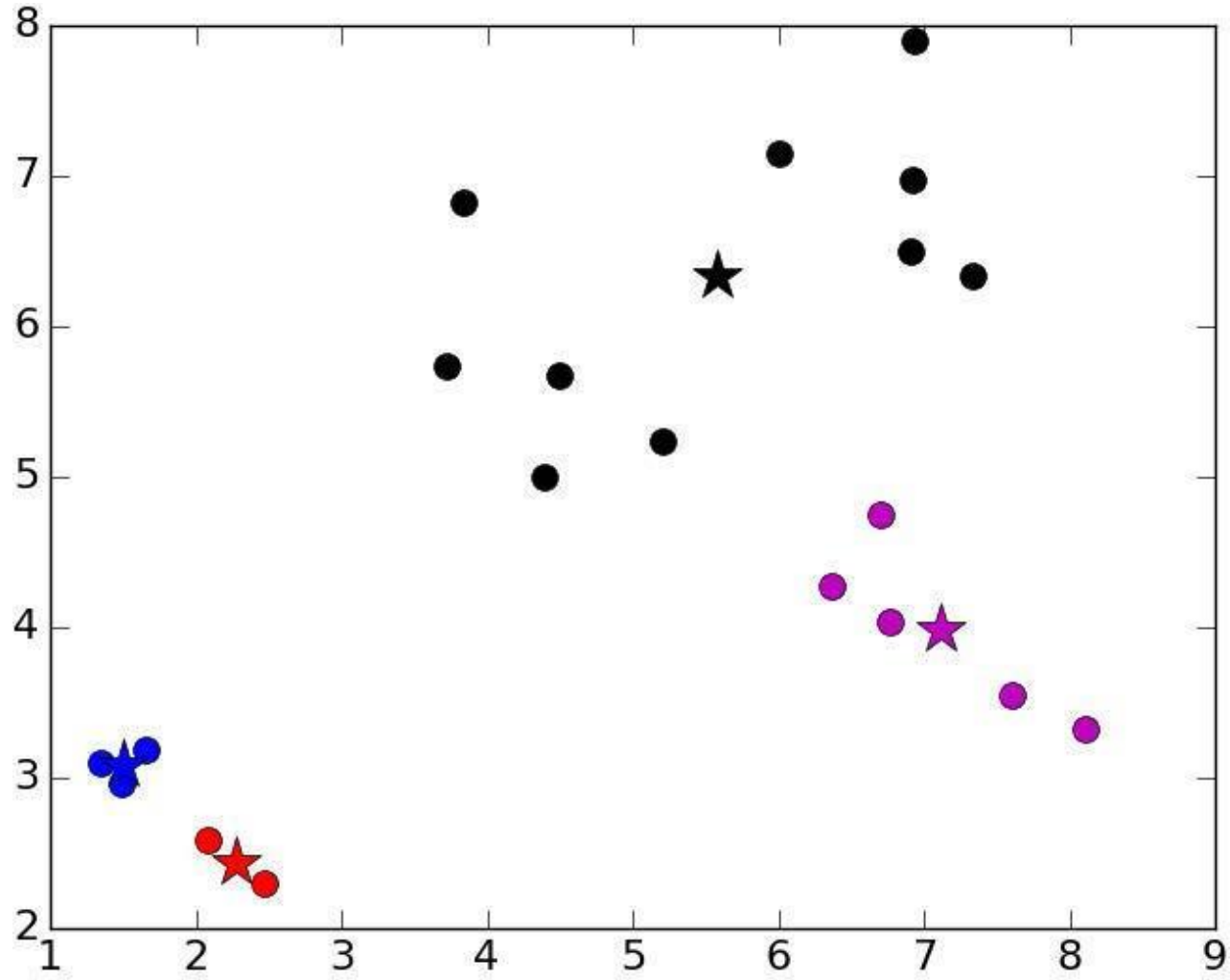
How to Choose K

- *A priori* knowledge about application domain
  - There are two kinds of people in the world: k = 2
  - There are five different types of bacteria: k = 5
- Search for a good k
  - Try different values of k and evaluate quality of results
  - Run hierarchical clustering on subset of data

Unlucky Initial Centroids



Converges On



Mitigating Dependence on Initial Centroids

Try multiple sets of randomly chosen initial centroids

Select “best” result

```
best = kMeans(points)
for t in range(numTrials):
   C = kMeans(points)
   if dissimilarity(C) < dissimilarity(best):
      best = C
return best
```

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

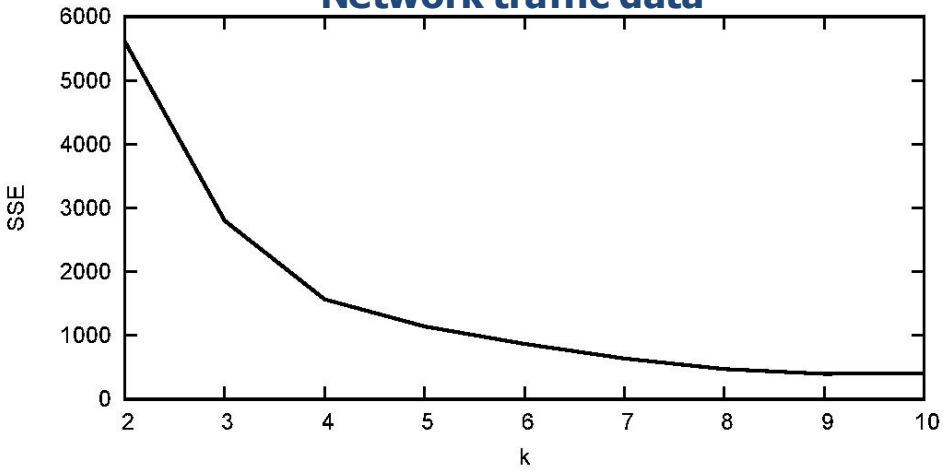
From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

K-means parameter setting

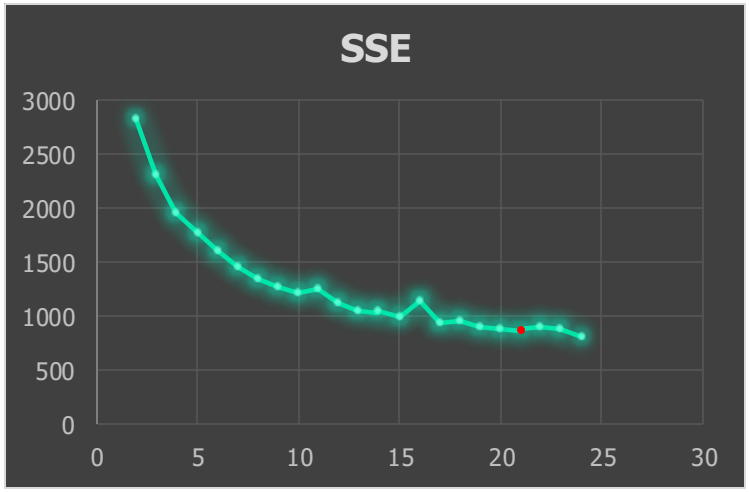
Elbow graph (Knee approach)

- Plotting the quality measure trend (e.g., SSE) against K
- Choosing the value of K
  - the gain from adding a centroid is negligible
  - The reduction of the quality measure is not interesting anymore

Network traffic data



Medical records



Recap from past lecture

1. Types of validations: leave-one-out, and 10-fold cross validations
2. Supervised vs. unsupervised: it’s about the data, isn’t ;)

DBSCAN

DBSCAN is a density-based algorithm

Density = number of points within a specified radius (Eps)

A point is a core point if it has more than a specified number of points (MinPts) within Eps

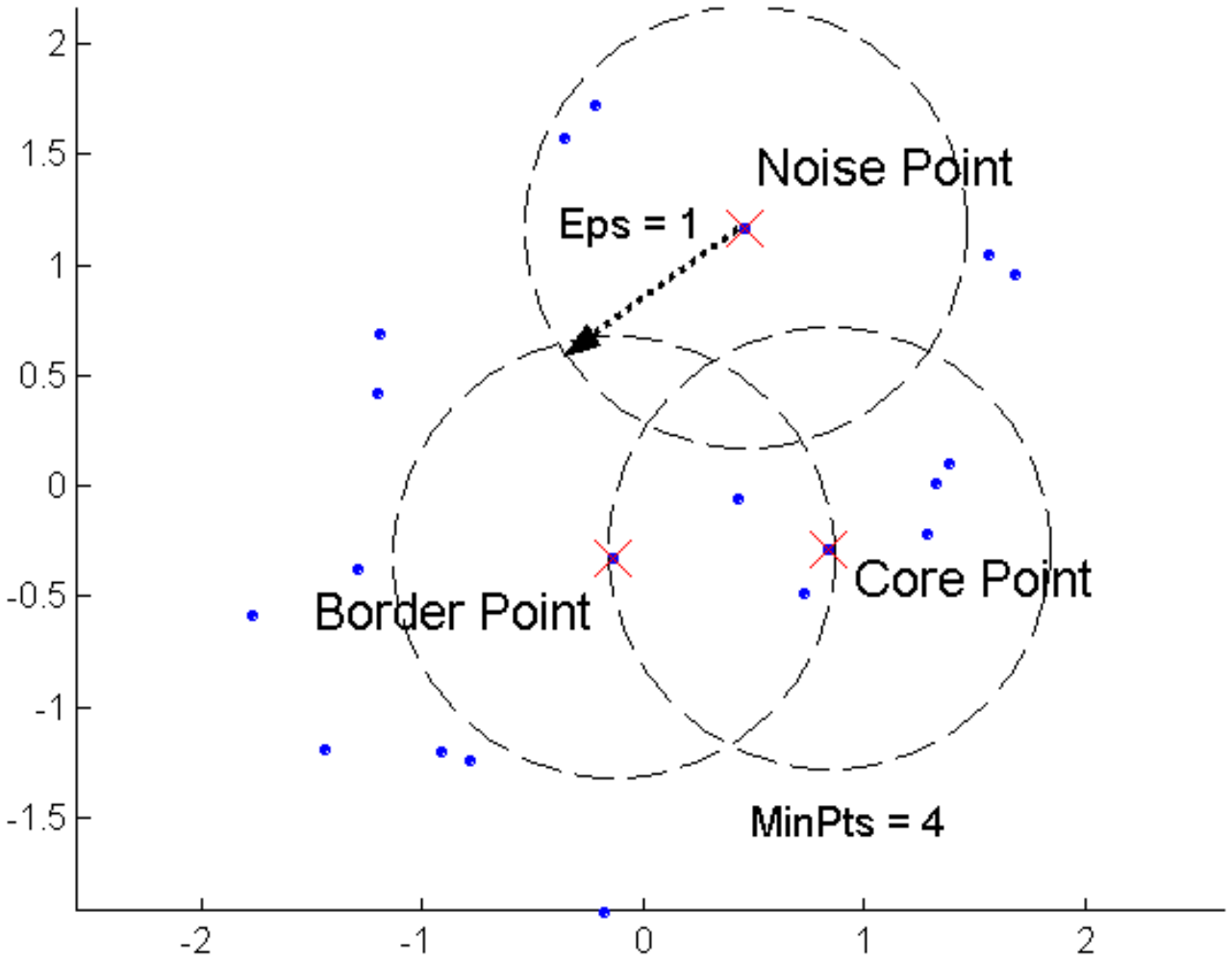
These are points that are at the interior of a cluster

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.

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

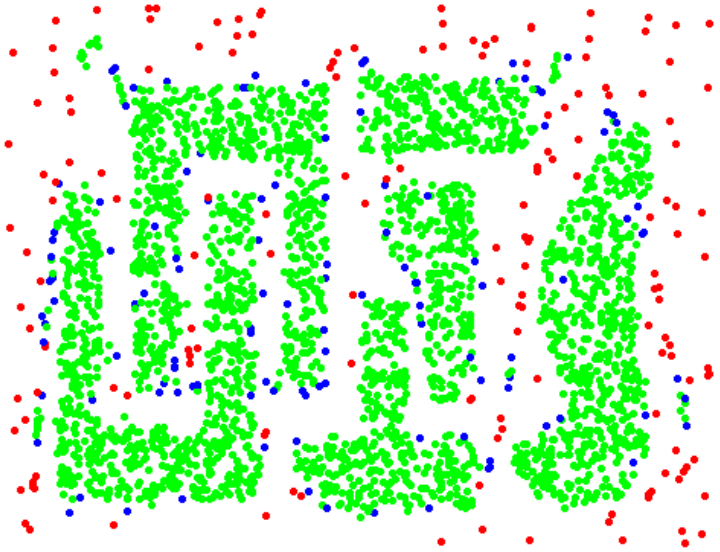
DBSCAN: Core, Border, and Noise Points



From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

DBSCAN: Core, Border, and Noise Points

Original Points



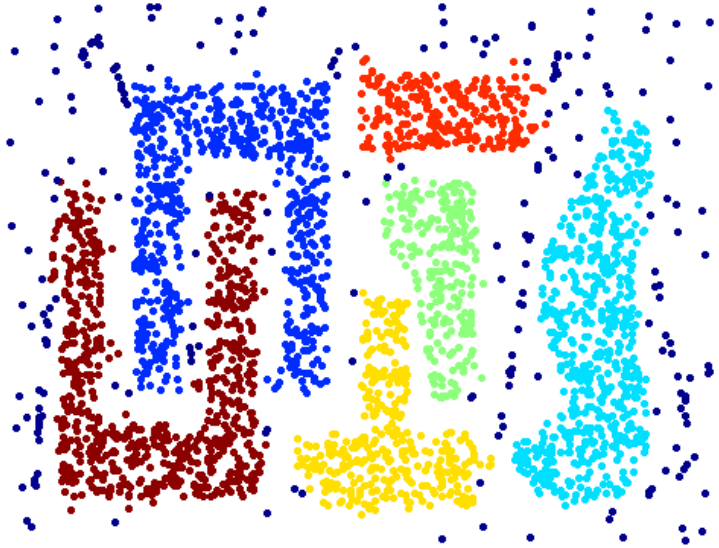
Point types: core, border and noise

Eps = 10, MinPts = 4

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

When DBSCAN Works Well

Original Points

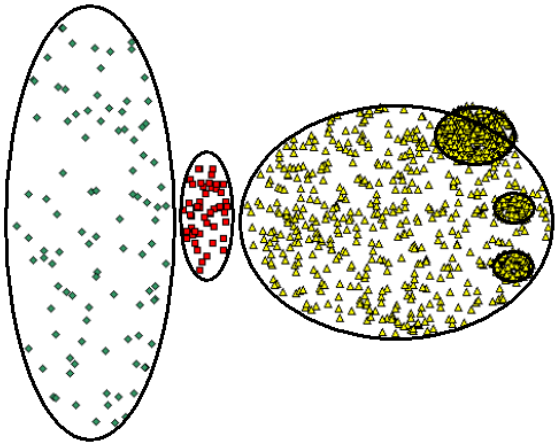


Clusters

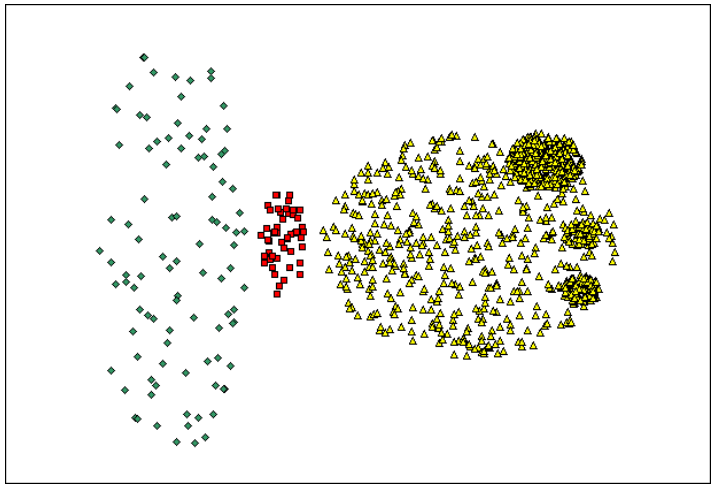
- Resistant to Noise
- Can handle clusters of different shapes and sizes

From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

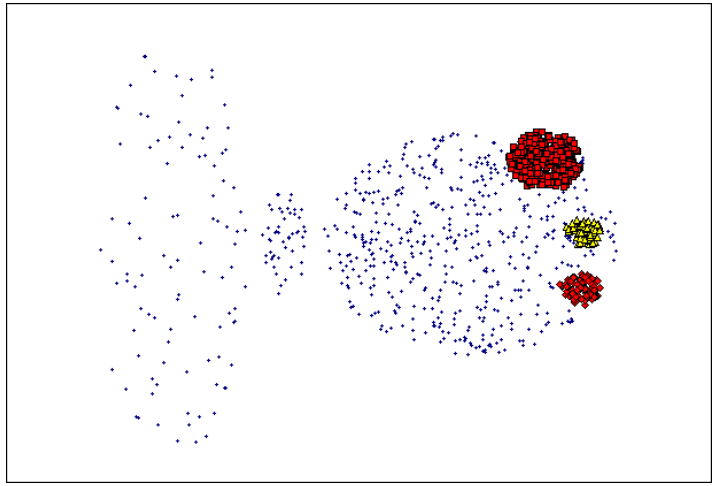
When DBSCAN Does NOT Work Well



Original Points



(MinPts=4, Eps=9.75).



(MinPts=4, Eps=9.62)

- Varying densities
- High-dimensional data

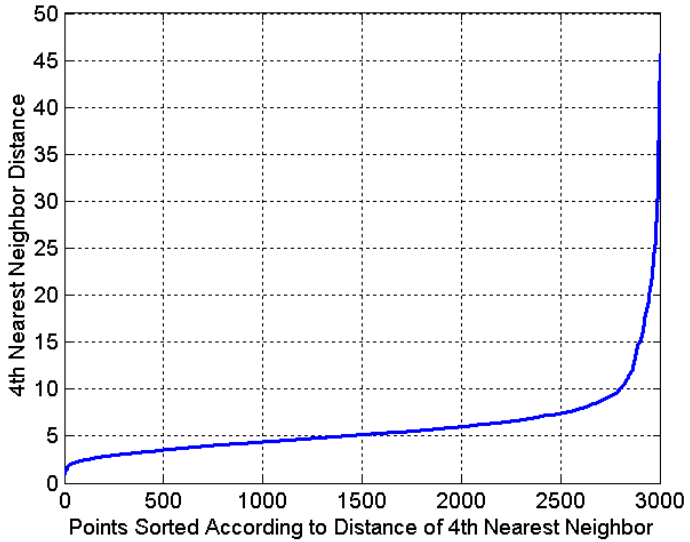
From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

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



From: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006

Internal measures: Silhouette

A succinct measure to evaluate how well each object lies within its cluster

It is defined for single points

It considers both cohesion and separation

Can be computed for

- Individual points
- Individual clusters
- Clustering result

Internal measures: Silhouette

For each object *i*

a(i): the average dissimilarity of *i* with all other objects within the same cluster (the smaller the value, the better the assignment)

b(i): min(average dissimilarity of *i* to any other cluster, of which *i* is not a member)

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

Ranges between -1 and +1

Typically between 0 and 1

The closer to 1, the better

Silhouette for clusters and clusterings

The average s(i) over all data of a *cluster* measures how tightly grouped all the data in the cluster are

The average s(i) over all data of the *dataset* measures how appropriately the data has been clustered