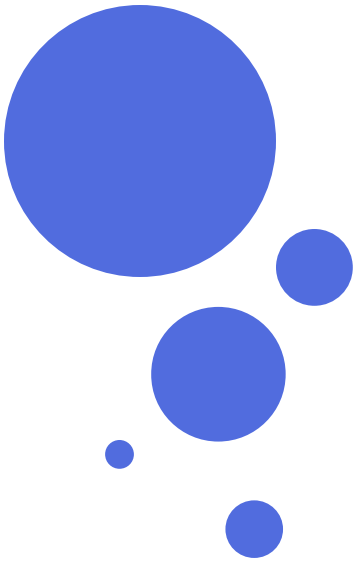
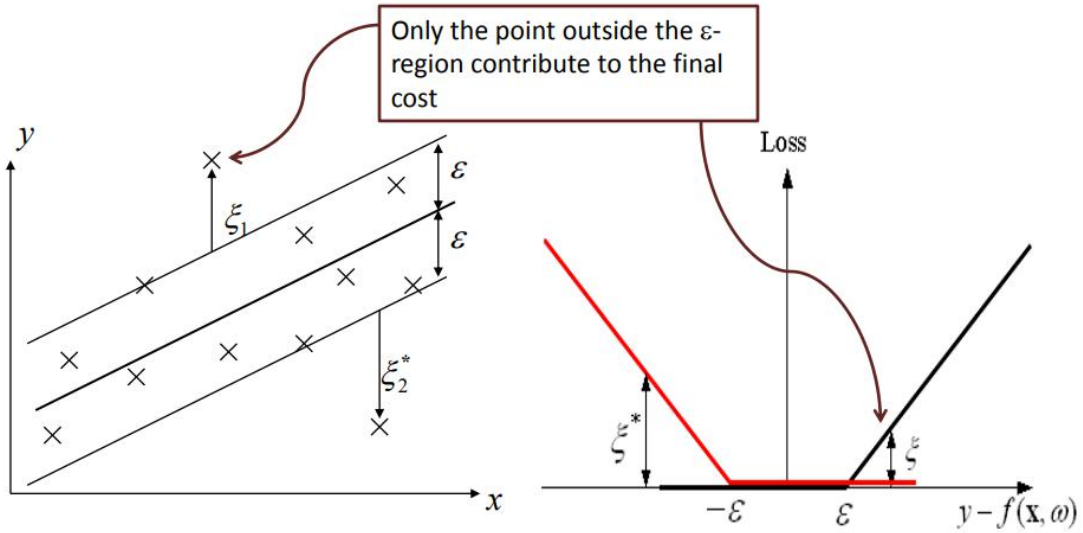


Lecture 21: Unsupervised Learning and Clustering Algorithms

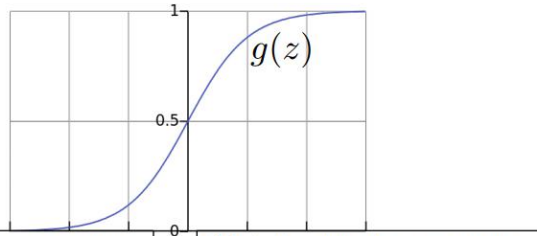
Dr. Chengjiang Long
Computer Vision Researcher at Kitware Inc.
Adjunct Professor at RPI.
Email: **longc3@rpi.edu**

Recap Previous Lecture



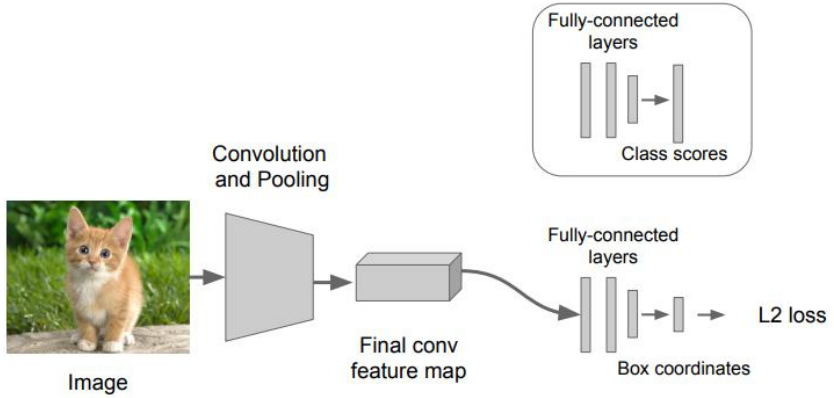
$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g\left(\boldsymbol{\theta}^{\mathsf{T}}\boldsymbol{x}\right)$$

$$g(z) = \frac{1}{1+e^{-z}}$$



$\boldsymbol{\theta}^{\mathsf{T}}\boldsymbol{x}$ should be large negative values for negative instances

$\boldsymbol{\theta}^{\mathsf{T}}\boldsymbol{x}$ should be large positive values for positive instances



Outline

- Introduce Unsupervised Learning and Clustering
- K-means Algorithm
- Hierarchy Clustering
- Applications

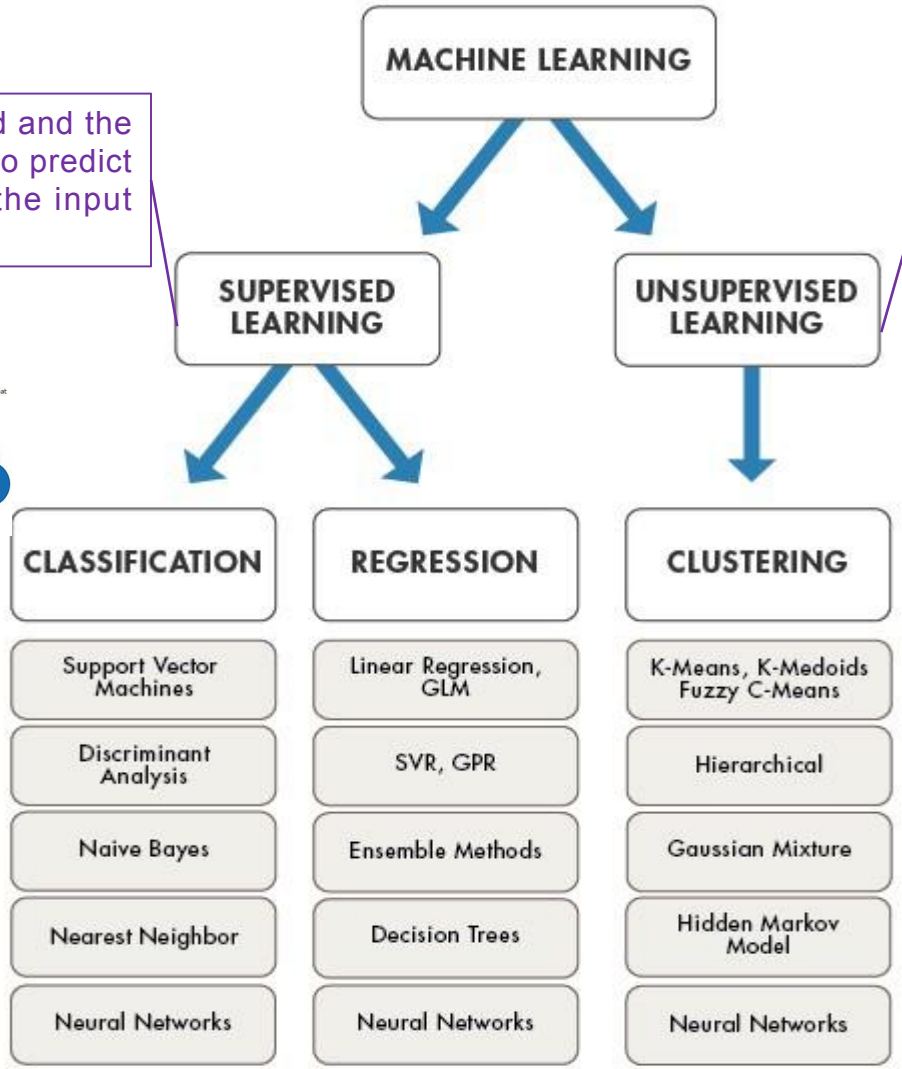
Outline

- **Introduce Unsupervised Learning and Clustering**
- K-means Algorithm
- Hierarchy Clustering
- Applications

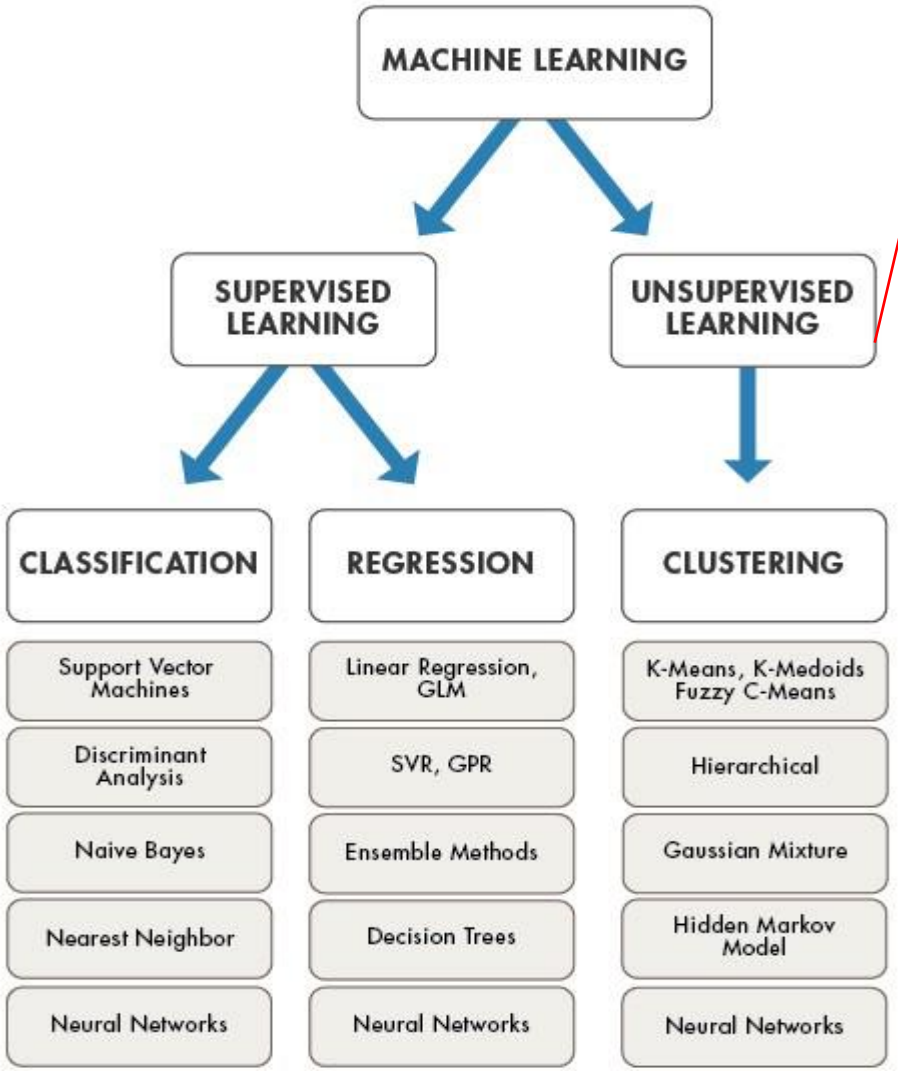
Unsupervised learning and clustering

All data is labeled and the algorithms learn to predict the output from the input data.

All data is unlabeled and the algorithms learn the inherent structure from the input data.



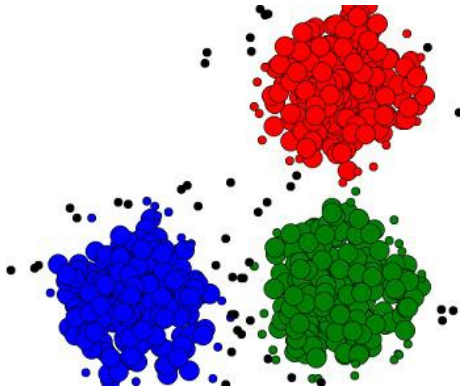
Unsupervised learning and clustering

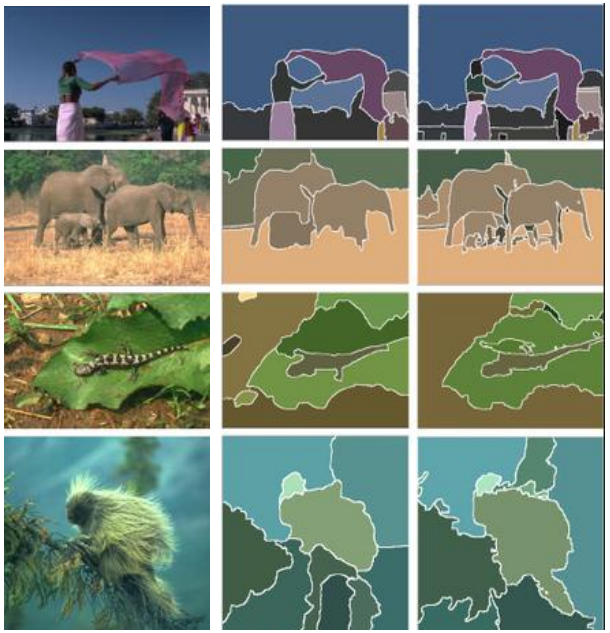


Goal: to model the underlying structure or distribution in the input data.

What is clustering?

- The organization of unlabeled data into similarity groups is called clustering.
- A cluster is a collection of data items which are "similar" between them, and "dissimilar" to data items into other clusters.
- Finding the class labels and the number of classes directly from the data (in contrast to classification)





What is clustering for?

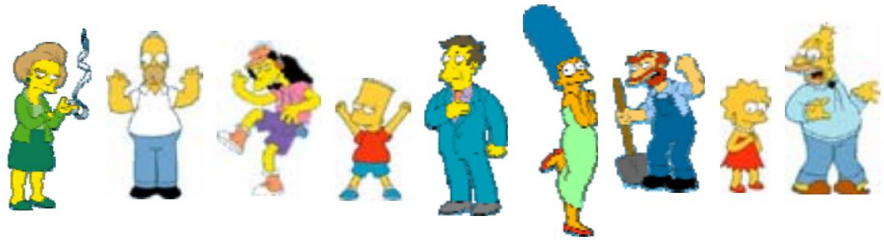
E.g.1: group people of similiar size together to make S,M,L T-shirts.

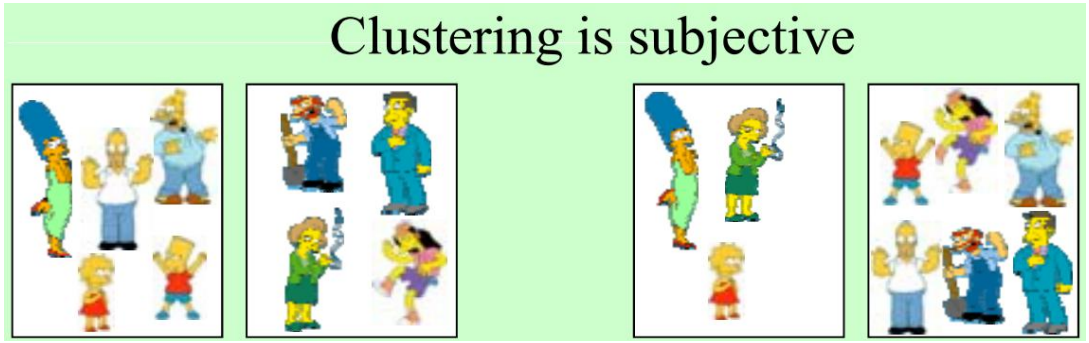
E.g.2: segment customers to do targeted marketing.

E.g.3: orgnize documents to produce a topic hierarchy.

What is clustering for?

- Clustering is one of the most utilized data mining techniques.
- Clustering is dependent on applications and to some extent is subjective.

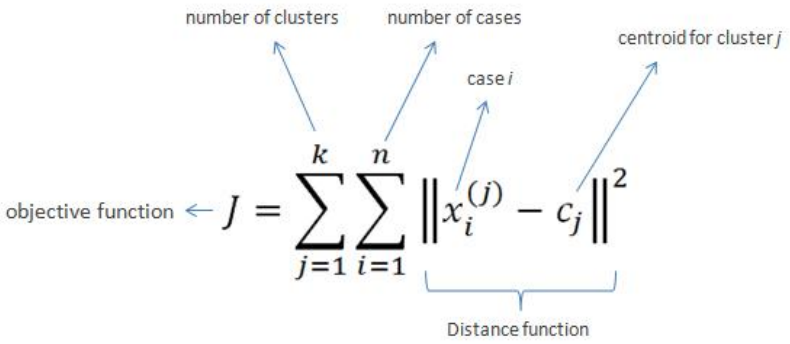


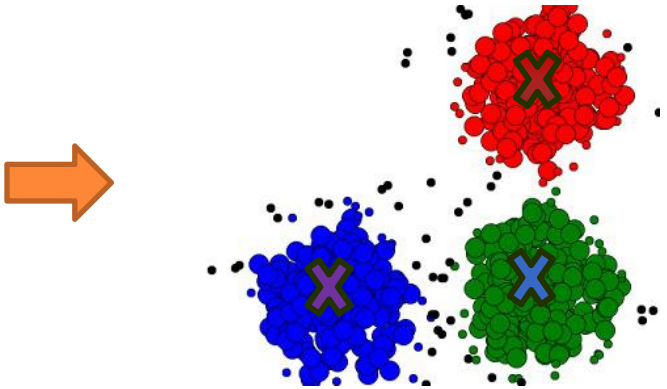


Clustering evaluation

- Intra-cluster cohesion (compactness)
  - Cohesion measures how near the data points in a cluster are to the cluster centroid.
  - Sum of squared error (SSE) is a commonly used measure.
- Inter-cluster seperation (isolation)
  - Separation means that different cluster centroids should be far away from one another.

Clustering is hard to evaluate. In most applications, expert judgements are still the key.





Data Clustering - Formal Definition

- Given a set of N unlabeled examples D = x_1, x_2, ..., x_N in a d-dimensional feature space, D is partitioned into a number of disjoint subsets Dj's:

$$D = \cup_{j=1}^{k} D_j \qquad D_i \cap D_j = \emptyset, i \neq j$$

- A partition is denoted by:

$$\pi = (D_1, D_2, ..., D_k)$$

and the problem of data clustering is thus formulated as

$$\pi^* = \underset{\pi}{\operatorname{argmin}}\ f(\pi)$$

where f(·) is formulated according to a given criterion.

Outline

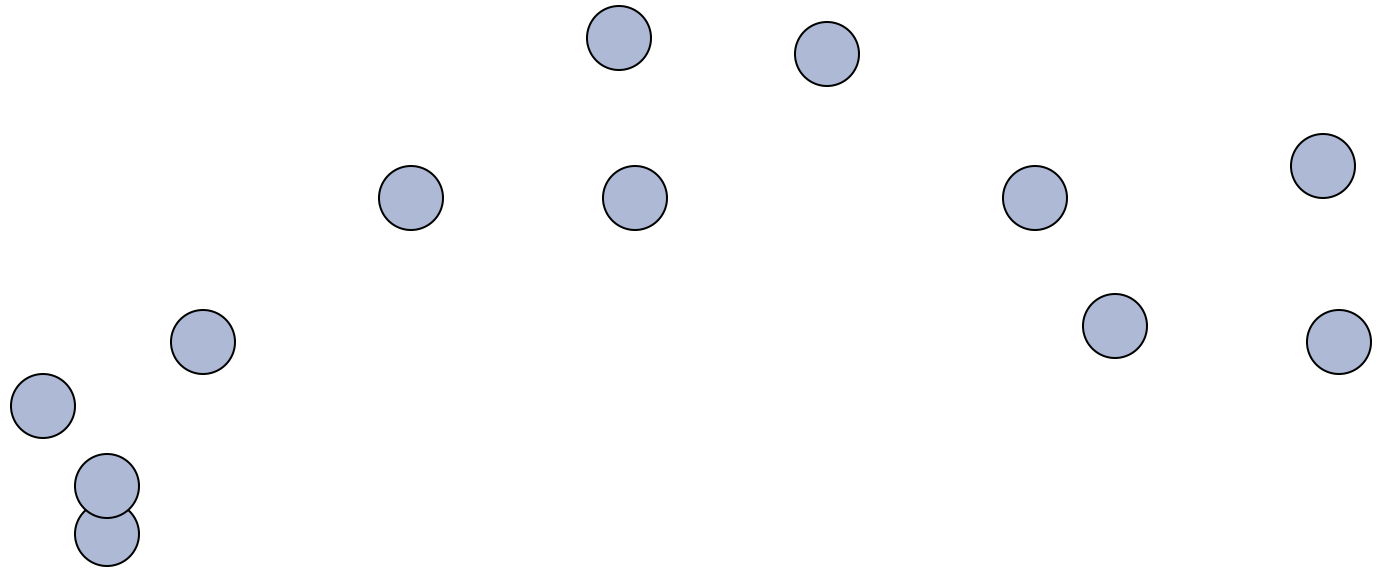
- Introduce Unsupervised Learning and Clustering
- **K-means Algorithm**
- Hierarchy Clustering
- Applications

K-means

- K-means is a most well-known and popular clustering algorithm.
- K-means procedure:
  - Start with some initial cluster centers
  - Iterate until there are no changes in any means
    - Assign/cluster each example to closest center
    - Recalculate centers as the mean of the points in a cluster

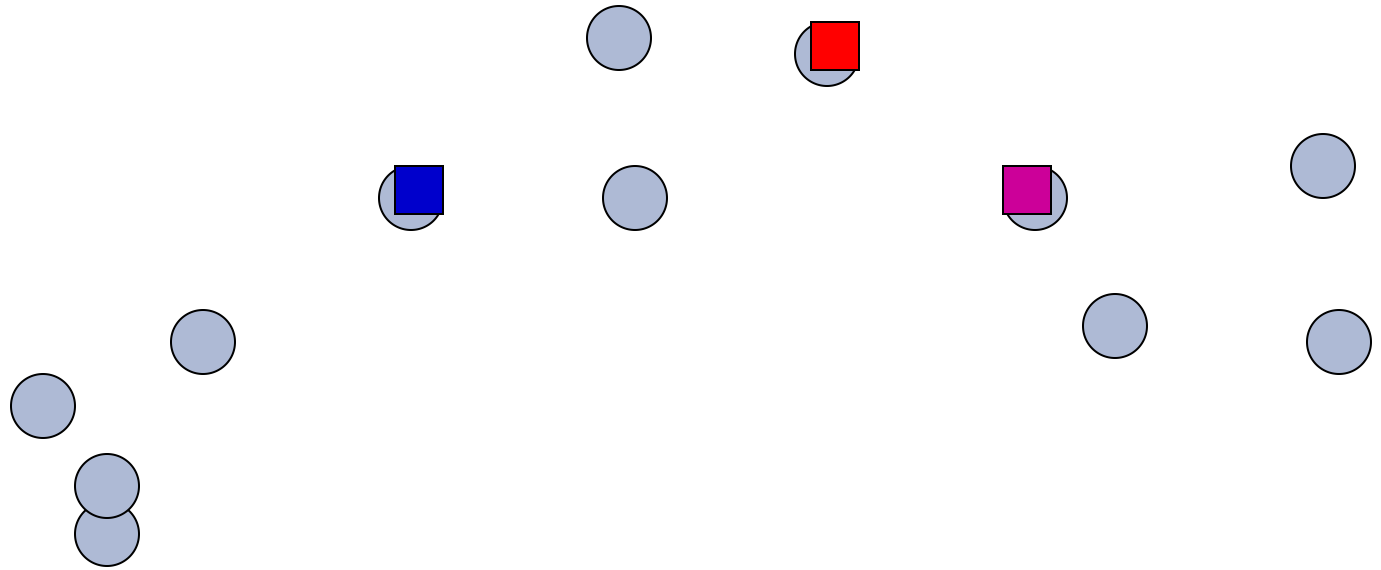
K-means: an example

- Start with some initial cluster centers
- Iterate until there are no changes in any means
  - Assign/cluster each example to closest center
  - Recalculate centers as the mean of the points in a cluster



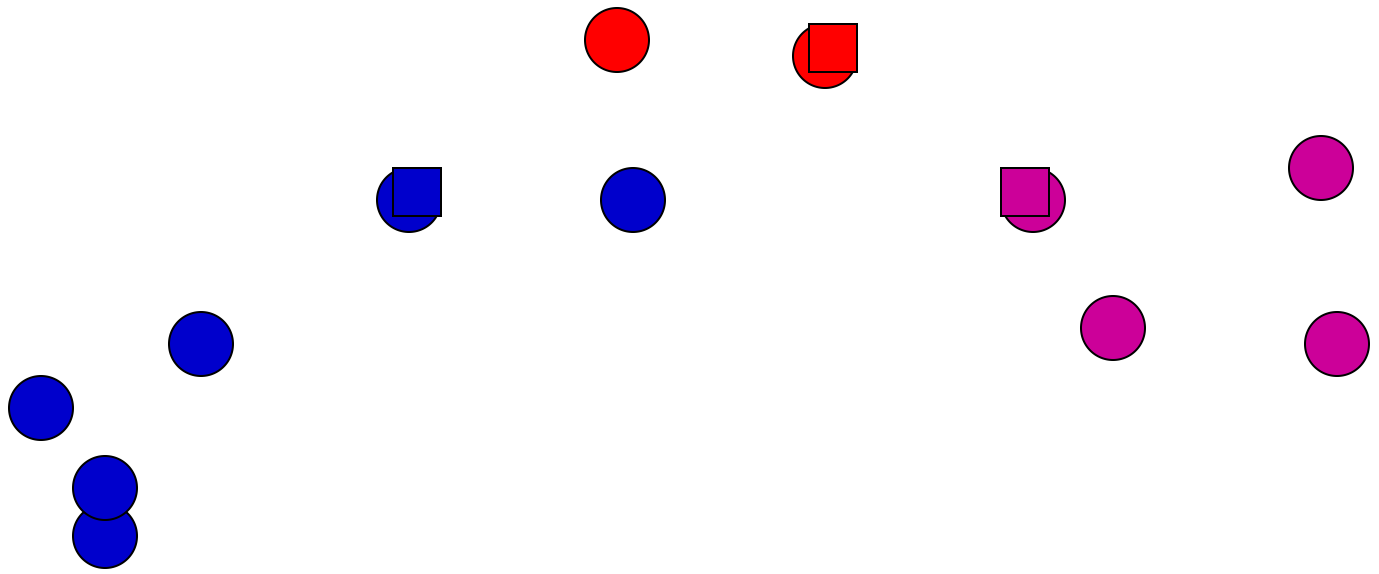
K-means: an example

- **Start with some initial cluster centers**
- Iterate until there are no changes in any means
  - Assign/cluster each example to closest center
  - Recalculate centers as the mean of the points in a cluster



K-means: an example

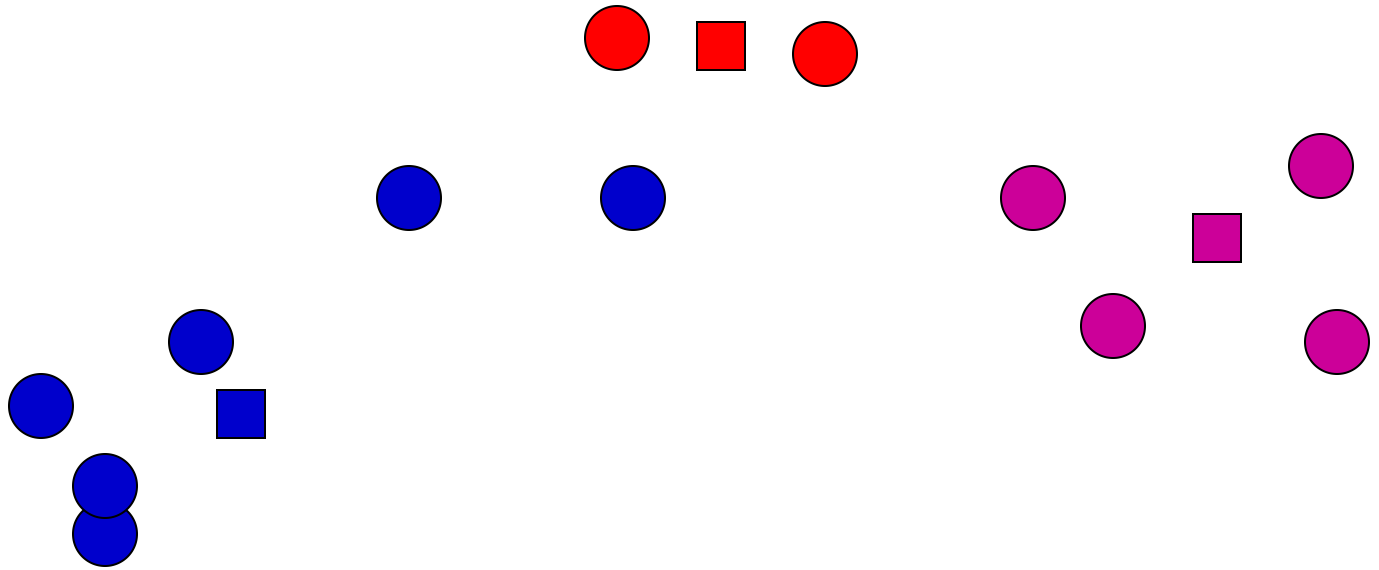
- Start with some initial cluster centers
- Iterate until there are no changes in any means
  - **Assign/cluster each example to closest center**
  - Recalculate centers as the mean of the points in a cluster



1–st iteration

K-means: an example

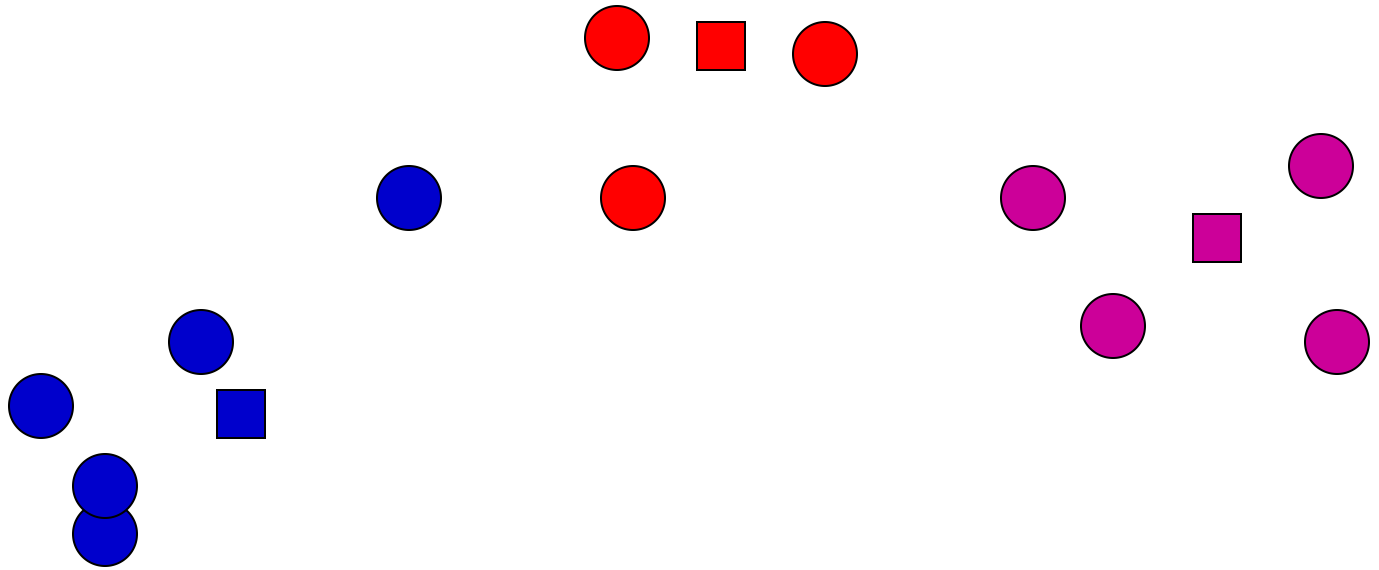
- Start with some initial cluster centers
- Iterate until there are no changes in any means
  - Assign/cluster each example to closest center
  - **Recalculate centers as the mean of the points in a cluster**



1–st iteration

K-means: an example

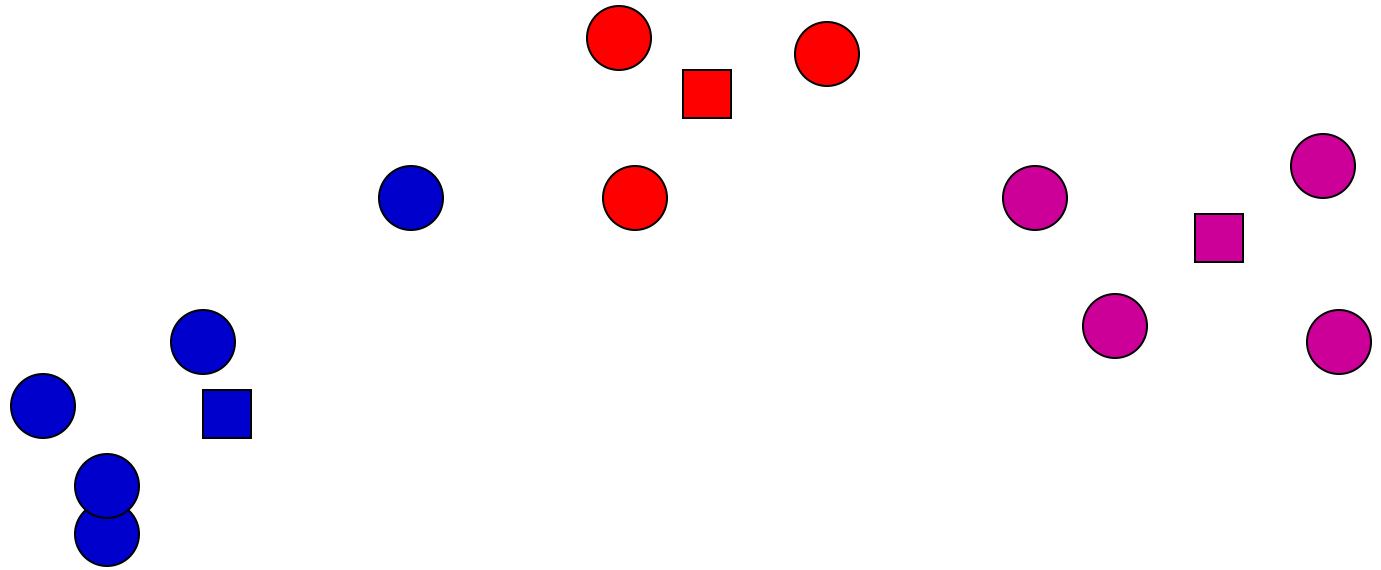
- Start with some initial cluster centers
- Iterate until there are no changes in any means
  - **Assign/cluster each example to closest center**
  - Recalculate centers as the mean of the points in a cluster



2–nd iteration

K-means: an example

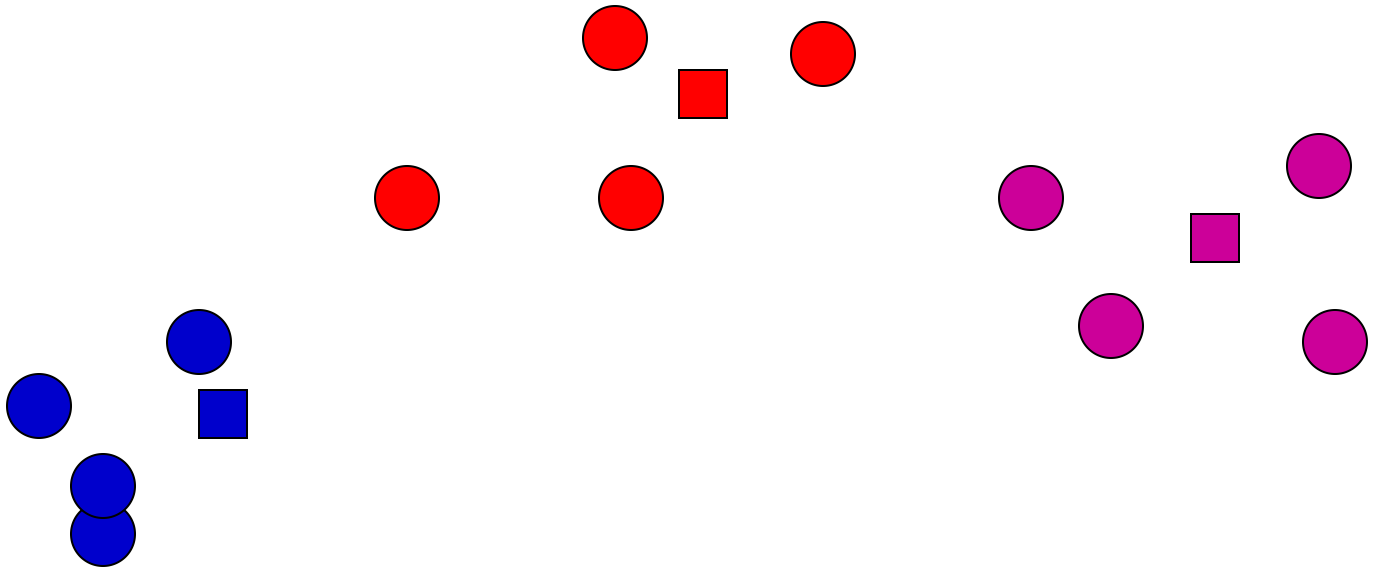
- Start with some initial cluster centers
- Iterate until there are no changes in any means
  - Assign/cluster each example to closest center
  - **Recalculate centers as the mean of the points in a cluster**



2–nd iteration

K-means: an example

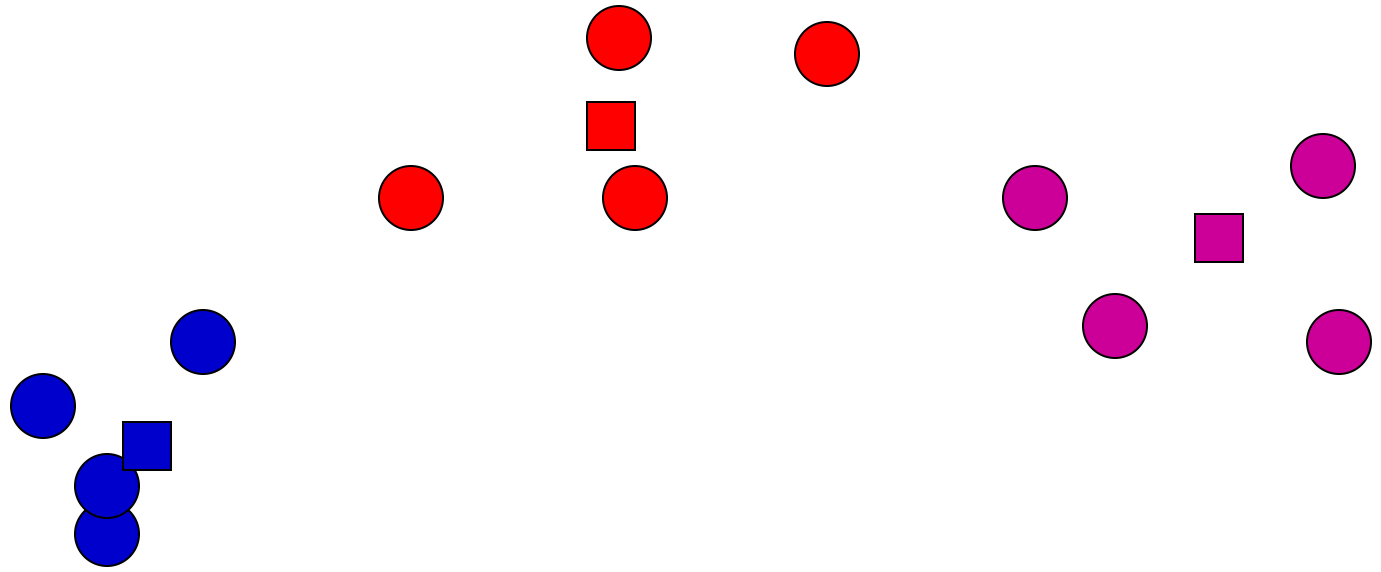
- Start with some initial cluster centers
- Iterate until there are no changes in any means
  - **Assign/cluster each example to closest center**
  - Recalculate centers as the mean of the points in a cluster



3–rd iteration

K-means: an example

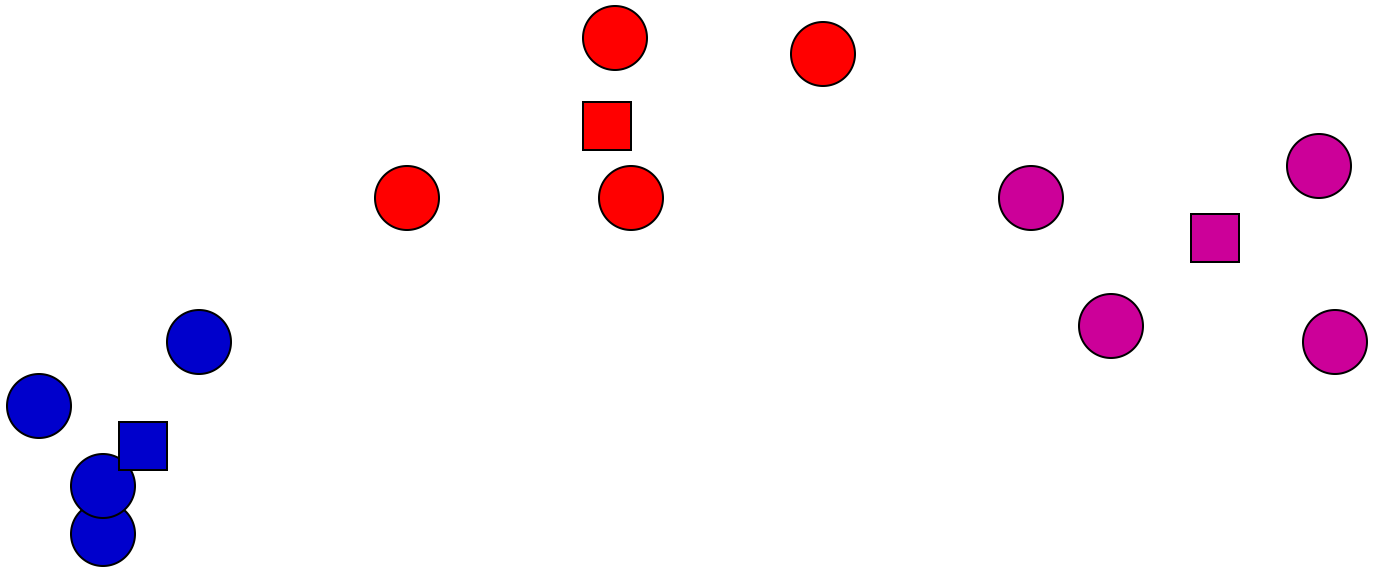
- Start with some initial cluster centers
- Iterate until there are no changes in any means
  - Assign/cluster each example to closest center
  - **Recalculate centers as the mean of the points in a cluster**



3–rd iteration

K-means: an example

- Start with some initial cluster centers
- **Iterate** until there are no changes in any means
  - Assign/cluster each example to closest center
  - Recalculate centers as the mean of the points in a cluster



No changes: Done

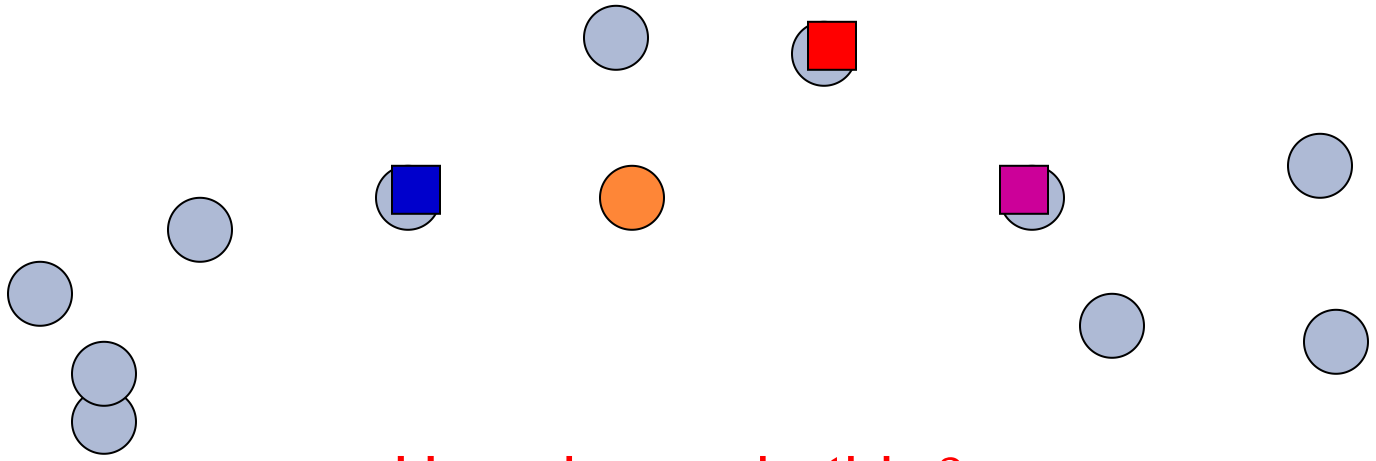
K-means

Iterate:

- **Assign/cluster each example to closest center**

  iterate over each point:
  - get **distance** to each cluster center
    - assign to closest center (hard cluster)

- Recalculate centers as the mean of the points in a cluster



How do we do this?

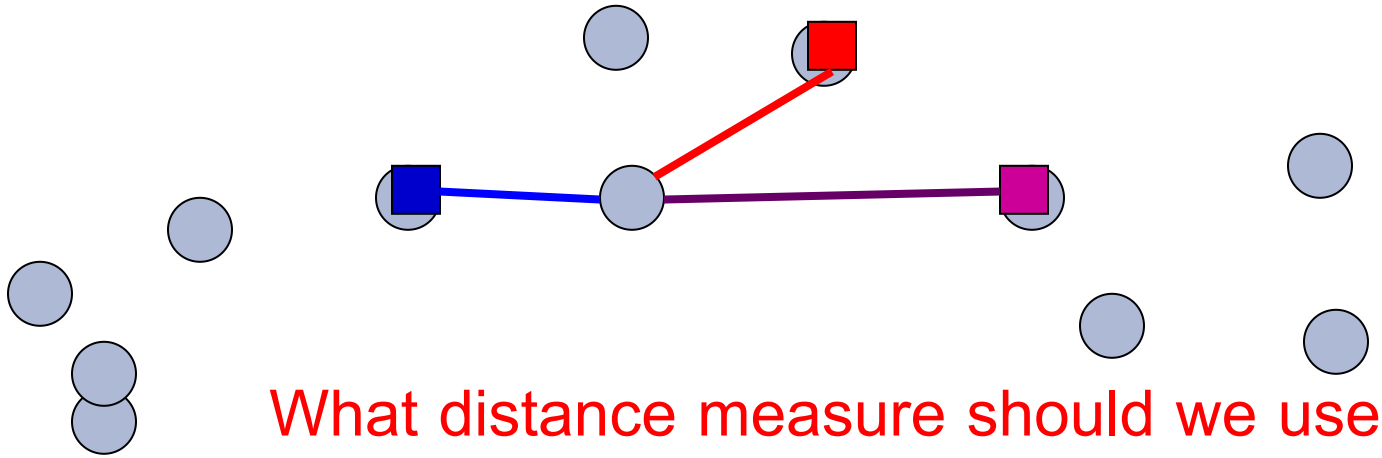
K-means

Iterate:

- **Assign/cluster each example to closest center**

  iterate over each point:
  - get **distance** to each cluster center
    - assign to closest center (hard cluster)

- Recalculate centers as the mean of the points in a cluster



What distance measure should we use?

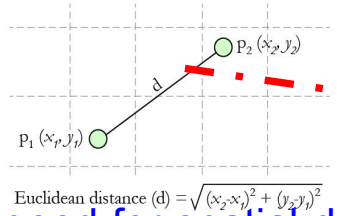
K-means

Iterate:

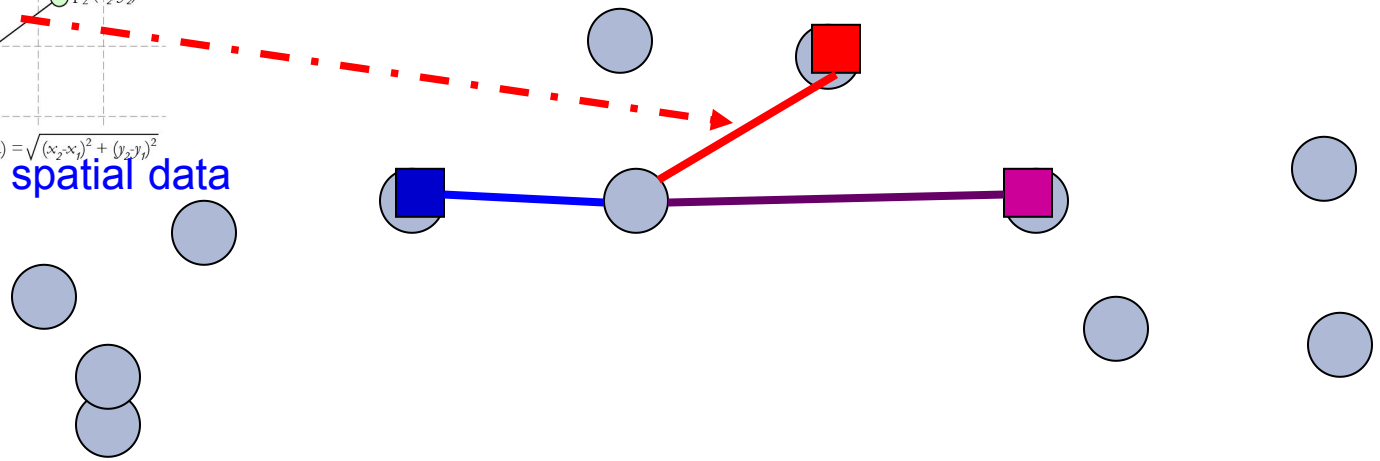
- **Assign/cluster each example to closest center**

  iterate over each point:
  get **distance** to each cluster center
  assign to closest center (hard cluster)

- Recalculate centers as the mean of the points in a cluster



good for spatial data



What distance measure should we use?

Euclidean Distance

- Euclidean Distance

$$dist = \sqrt{\sum_{k=1}^{n} (p_k - q_k)^2}$$

Where n is the number of dimensions (attributes) and p_k and q_k are, respectively, the kth attributes (components) or data objects p and q.

- Standardization is necessary, if scales differ.

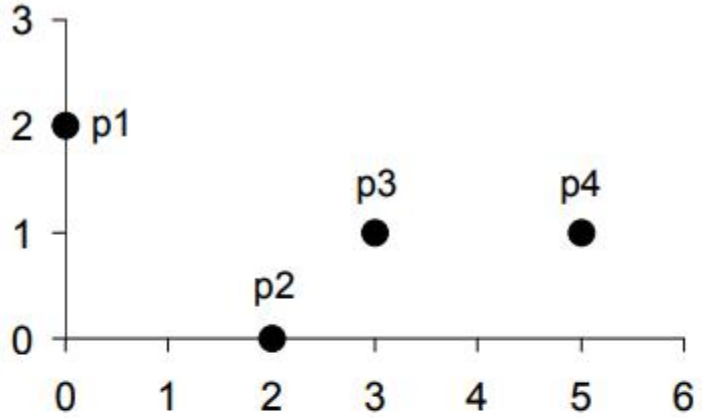
Minkowski Distance

- Minkowski Distance is a generalization of Euclidean Distance

$$dist = \left(\sum_{k=1}^{n} | p_k - q_k |^r \right)^{\frac{1}{r}}$$

Where r is a parameter, n is the number of dimensions (attributes) and p_k and q_k are, respectively, the k-th attributes (components) or data objects p and q.

Euclidean Distance



point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

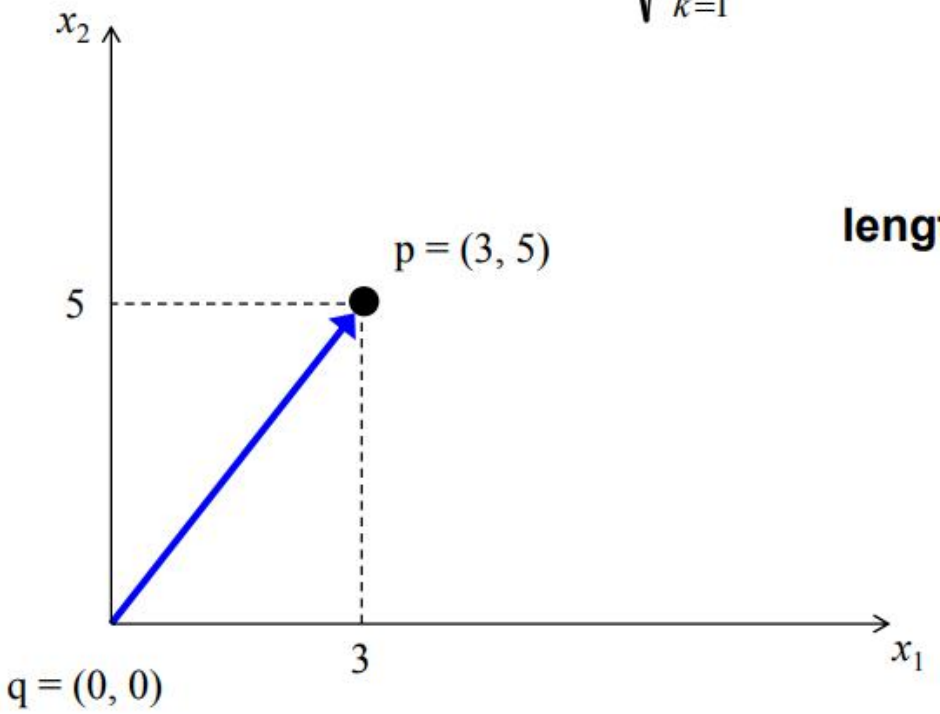
	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix

More about Euclidean distance

$$dist(p,q) = \sqrt{\sum_{k=1}^{n}(p_k - q_k)^2} = \sqrt{\sum_{k=1}^{n} p_k^2} = \|p\|$$

length of vector p



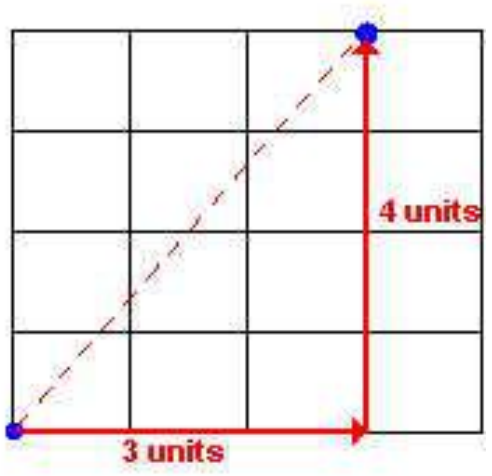
Manhattan Distance

- Manhattan distance represents distance that is measured along directions that are parallel to the x and y axes
- Manhattan distance between two *n*-dimensional vectors $x=(x_1,x_2,\ldots,x_n)$ and $y=(y_1,y_2,\ldots,y_n)$ is:

$$d_M(x,y) = |x_1 - y_1| + |x_2 - y_2| + \ldots + |x_n - y_n|$$

$$= \sum_{i=1}^{n} |x_i - y_i|$$

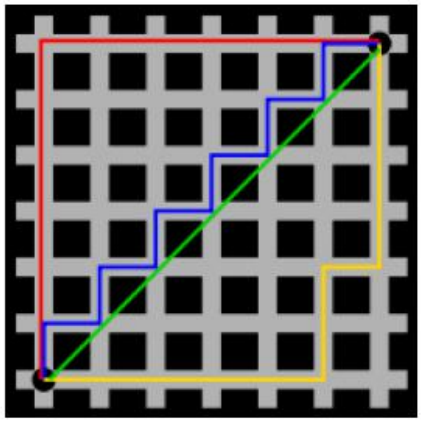
Where $|x_i - y_i|$ represents the absolute value of the difference betweeen x_i and y_i



Minkowski Distance: Examples

- $r = 1$. City block (Manhattan, taxicab, L_1 norm) distance.
  - A common example of this is the Hamming distance, which is just the number of bits that are different between two binary vectors
- $r = 2$. Euclidean distance
- $r \rightarrow \infty$. "supremum" (L_{max} norm, L_∞ norm) distance.
  - This is the maximum difference between any component of the vectors

- Do not confuse r with n, i.e., all these distances are defined for all numbers of dimensions.



From Wikipedia

Minkowski Distance

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

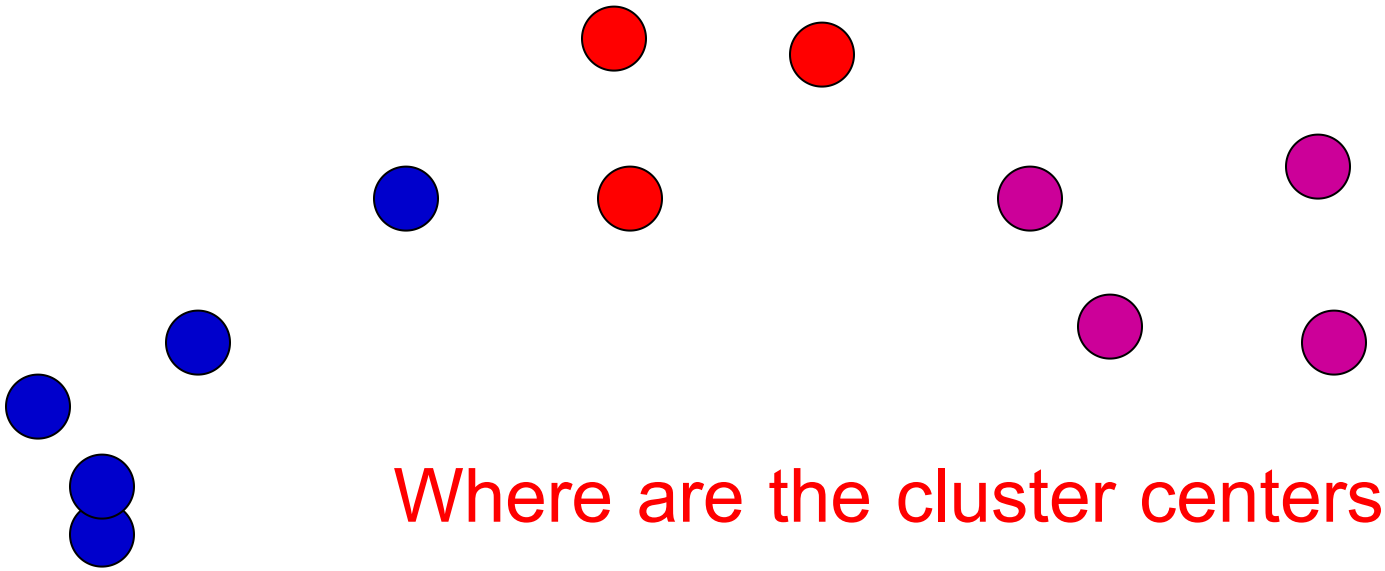
L_∞	p1	p2	p3	p4
p1	0	2	3	5
p2	2	0	1	3
p3	3	1	0	2
p4	5	3	2	0

Distance Matrix

K-means

Iterate:

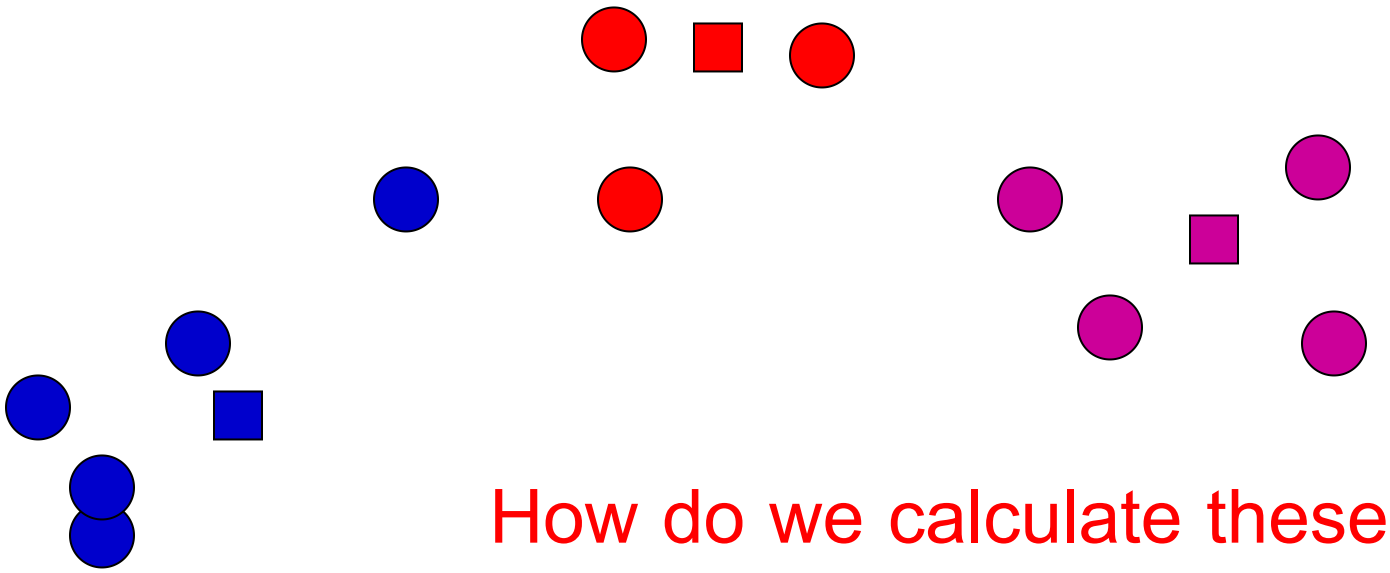
- Assign/cluster each example to closest center
- Recalculate centers as the mean of the points in a cluster



K-means

Iterate:

- Assign/cluster each example to closest center
- Recalculate centers as the mean of the points in a cluster



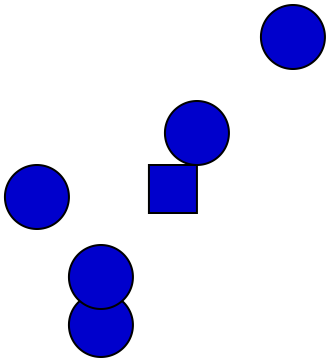
How do we calculate these?

K-means

Iterate:

- Assign/cluster each example to closest center
- Recalculate centers as the mean of the points in a cluster

Mean of the points in the cluster:



$$C_x = \frac{1}{|C|} \sum_{i=1}^{|C|} x_i$$

$$C_y = \frac{1}{|C|} \sum_{i=1}^{|C|} y_i$$

Pros and cons of K-means

Strengths:

- **Simple**: easy to understand and to implement.
- **Efficient**: time complexity is $O(tkn) \approx O(n)$.
  - n is the number of data point, k is the number of clusters, and t is the number of iterations.
  - Since both k and t are small, k-means is considered a linear algorithm.

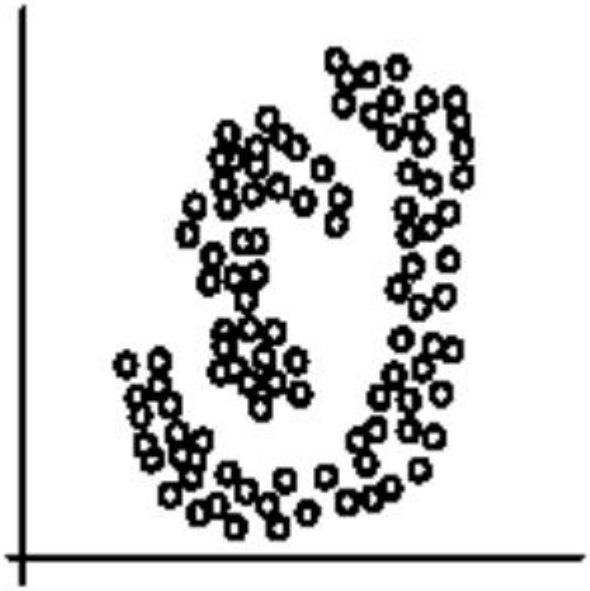
Note that: *it terminates at a local optimum if SSE is used. The global optimum is hard to find due to complexity*.

Weakneses:

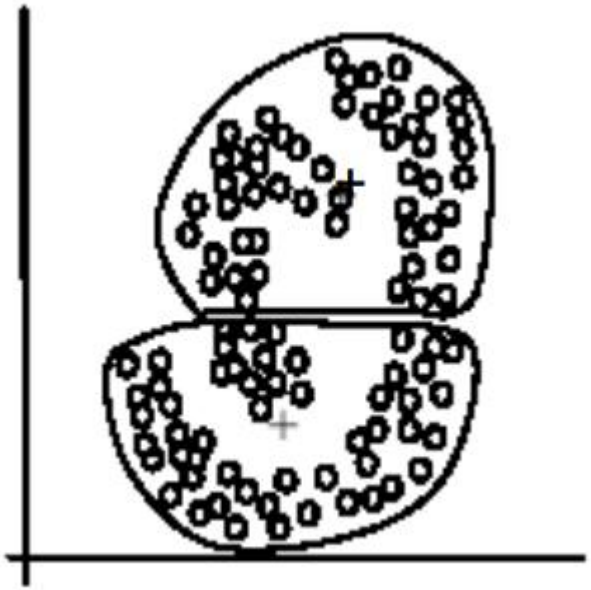
- The user needs to specify the value of K.
- Applicable only when mean is defined.
- The algorithm is sensitive to the initial seeds.
- The algorithm is sensitive to outliers.
  - Outliers are data points that are very far away from other data points.
  - Outliers could be errors in the data recording or some special data points with very different values.

Failure case

The k-means algorithm is not suitable for discovering clusters that are not hyper-ellipsoids (or hyper-spheres).

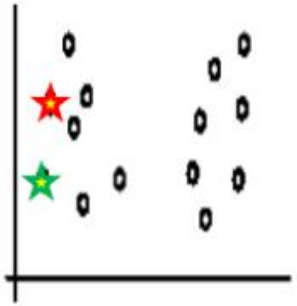


(A): Two natural clusters

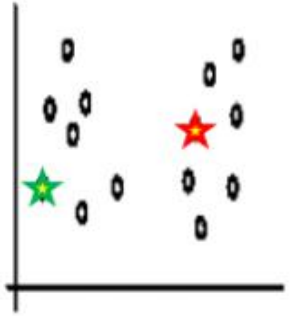


(B): *k*-means clusters

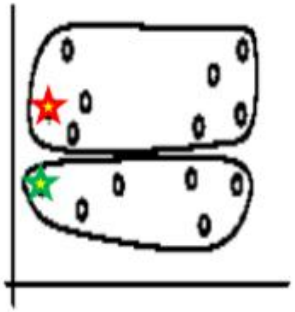
Sensitive to initial seeds



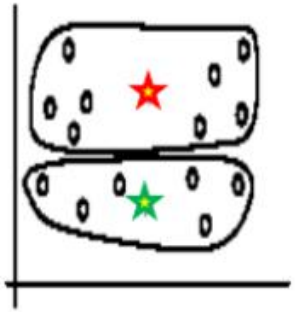
Random selection of seeds (centroids)



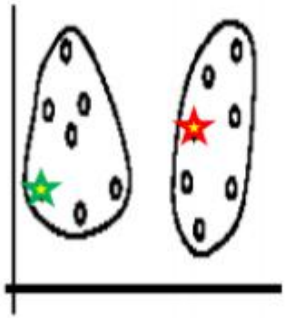
Random selection of seeds (centroids)



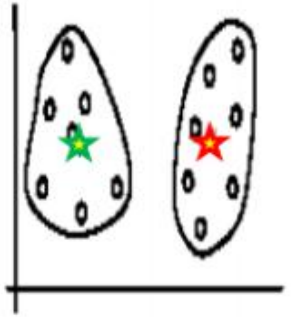
Iteration 1



Iteration 2

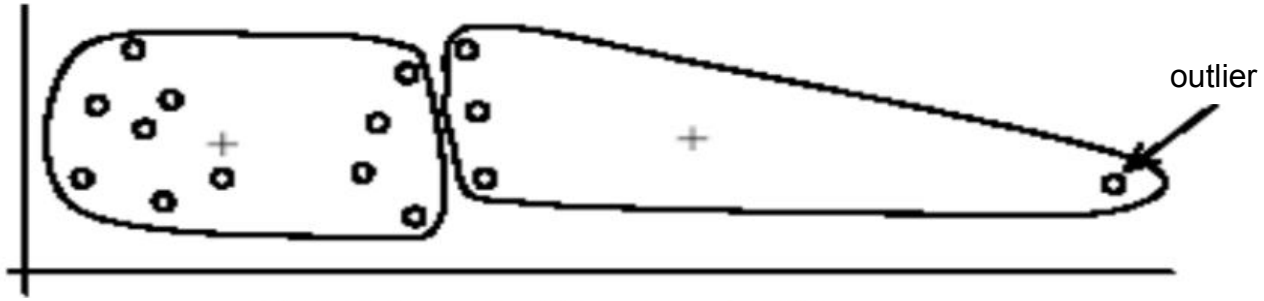


Iteration 1

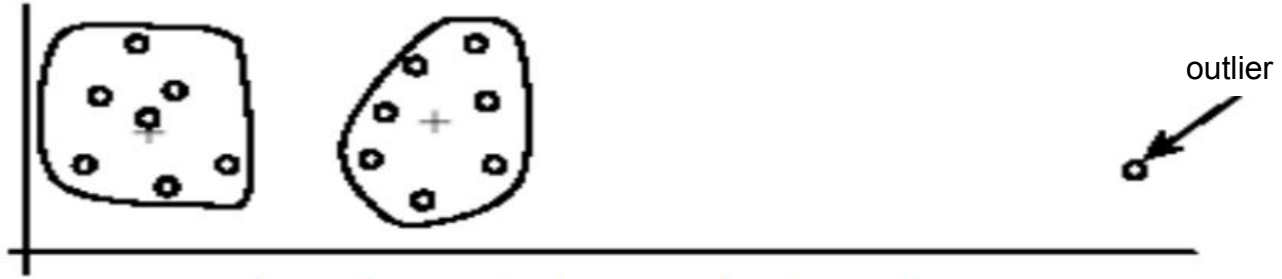


Iteration 2

Sensitive to outliers



Clustering results of K–means algorithm

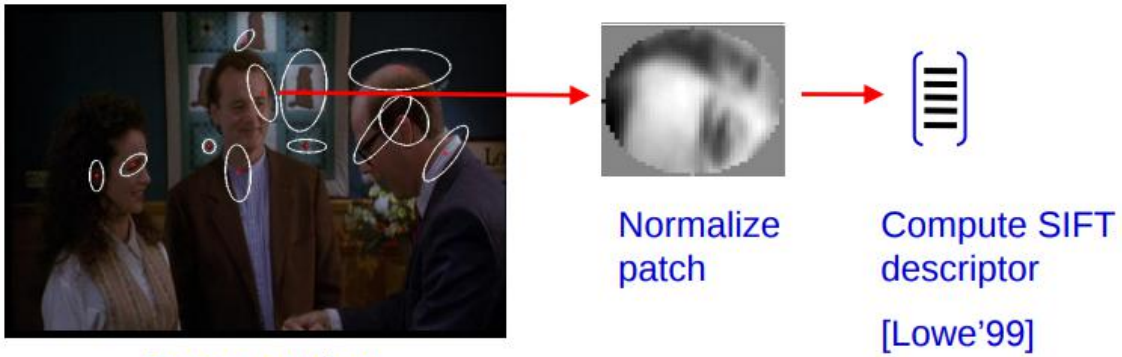


Clustering result after removing the outlier

- ✓ Remove some data points that are much further away the centroids than other data points.
- ✓ Perform random sampling: by choosing a small subset of the data points, the chance of selecting an outlier is much smaller.

Application to visual object recognition: Bag of Words

Application to visual object recognition: Bag of Words



Vector quantize descriptors from a set of training images using k/ means

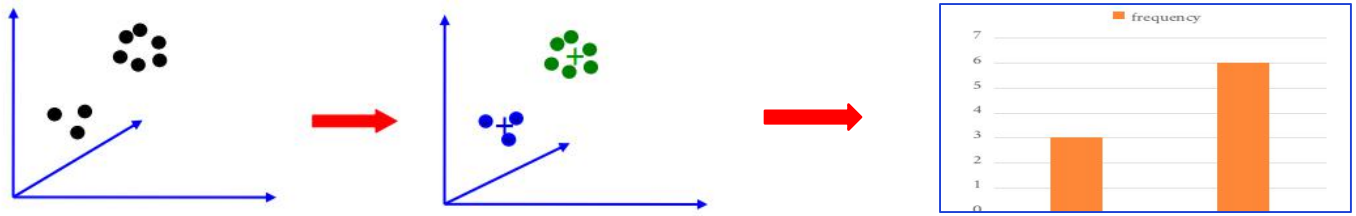
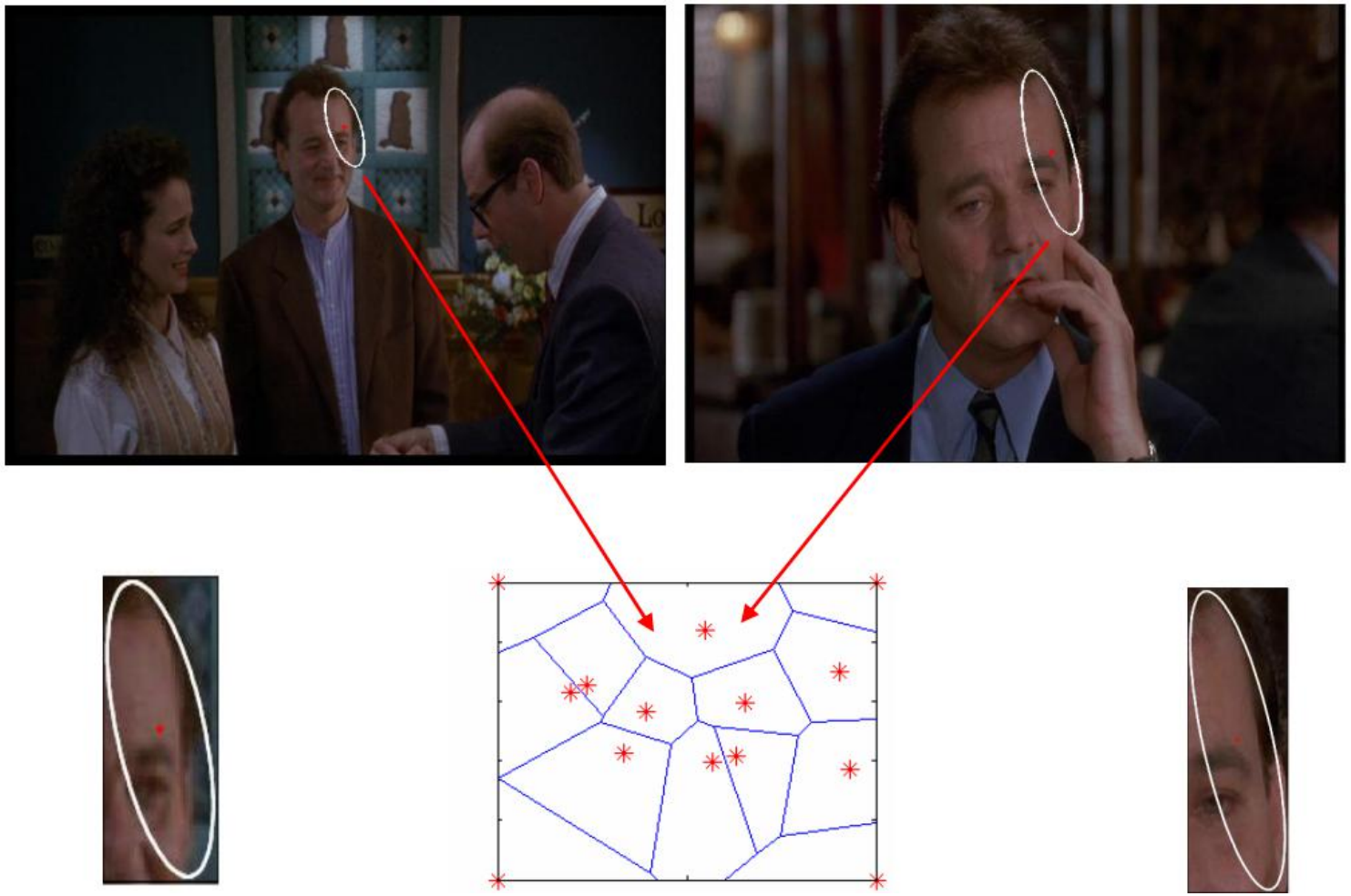


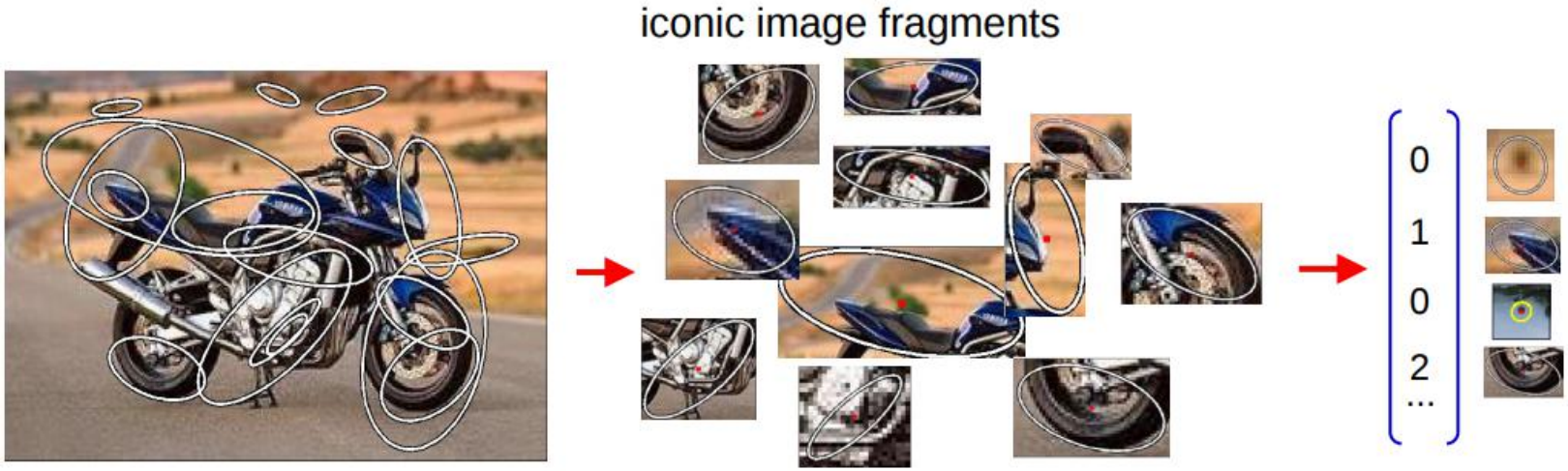
Image representation: a normalized histogram of visual words.

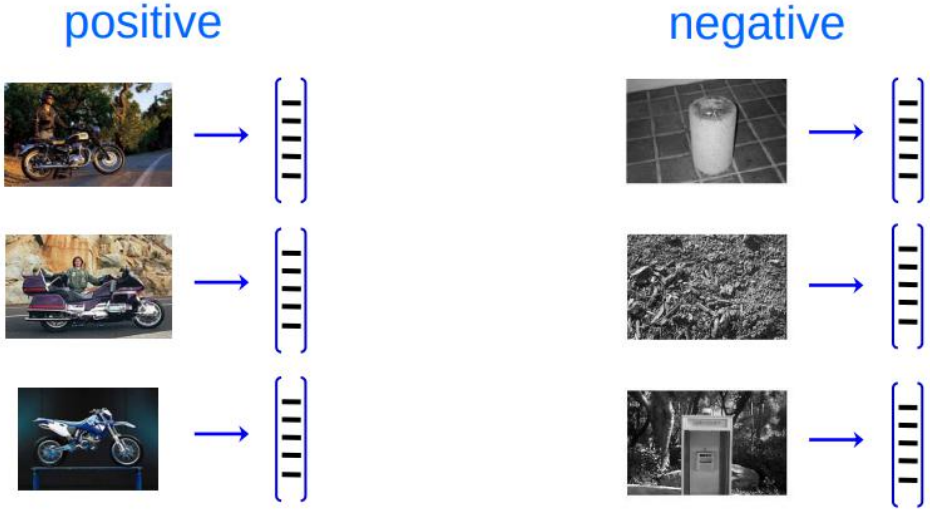
Application to visual object recognition: Bag of Words



The same visual word

Application to visual object recognition: Bag of Words





Summary

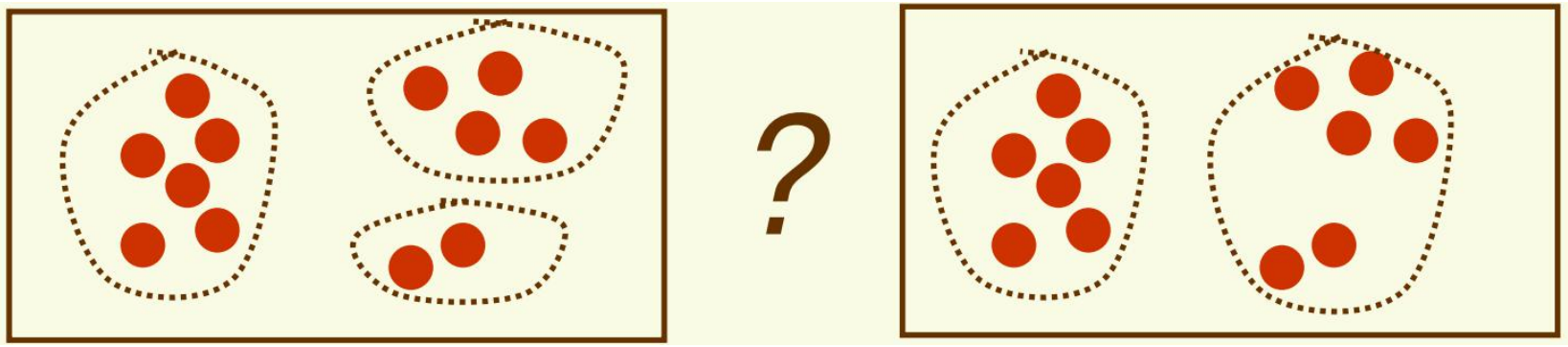
- ✓ Clustering is one of the most utilized data mining techniques.
- ✓ Clustering is highly application dependent and to some extent is subjective.
- ✓ Clustering is hard to evaluate, but very useful in practice.
- ✓ K-means algorithms is simple and efficient, but to sensitive to initial seeds and outliers. There are several improved versions of the original K-means algorithm, e.g., K-means++ algorithm.

Outline

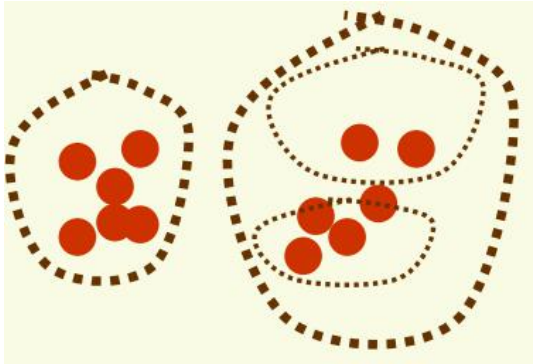
- Introduce Unsupervised Learning and Clustering
- K-means Algorithm
- **Hierarchy Clustering**
- Applications

Hierarchical Clustering

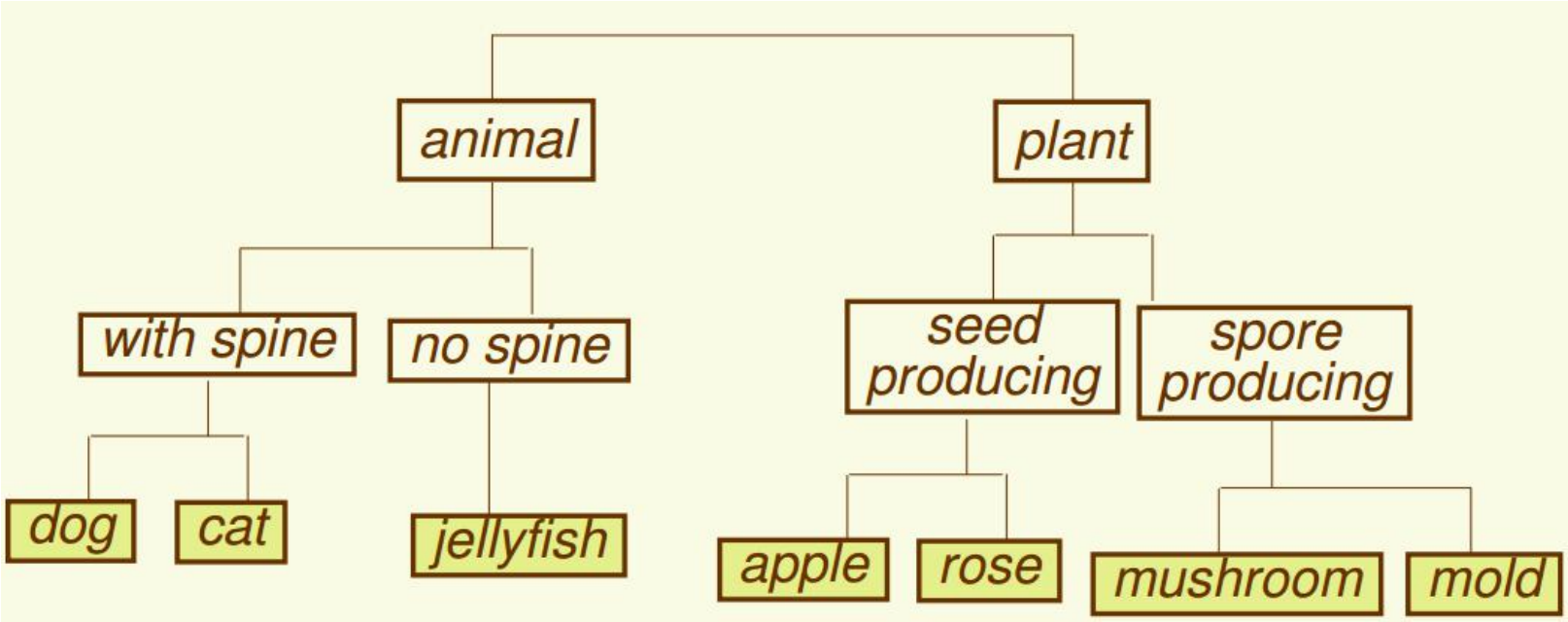
- Up to now, considered “flat” clustering



- For some data, hierarchical clustering is more appropriate than “flat” clustering
- Hierarchical clustering

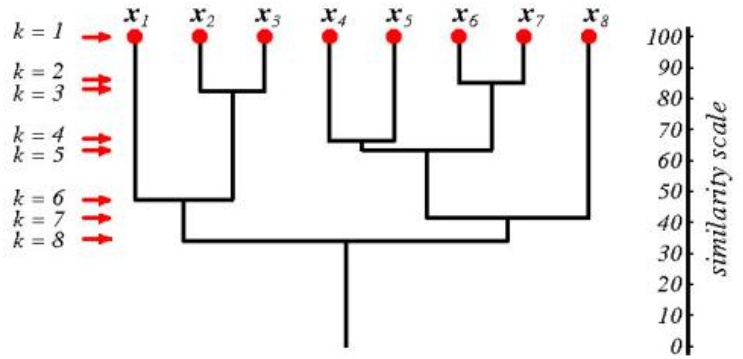


Hierarchical Clustering: Biological Taxonomy



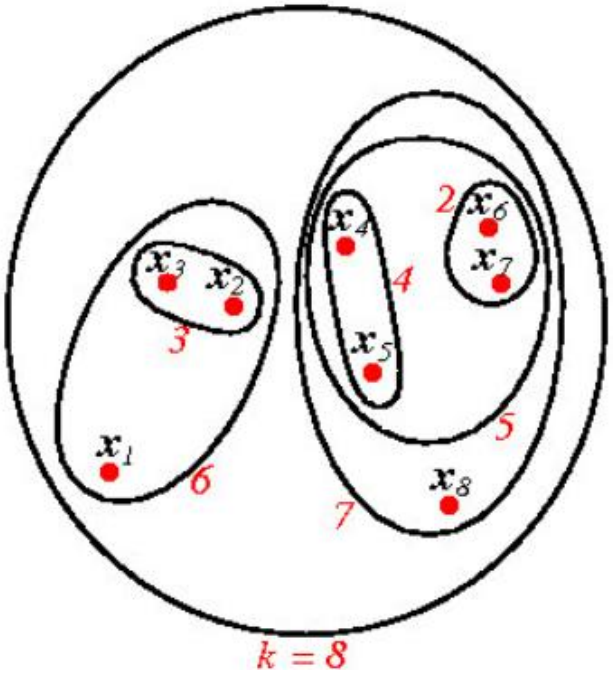
Hierarchical Clustering: Dendogram

- Prefered way to represent a hierarchical clustering is a dendogram
  - _x0001_ Binary tree
  - _x0001_ Level k corresponds to partitioning with n-k+1 clusters
  - _x0001_ if need k clusters, take clustering from level n-k+1
  - _x0001_ If samples are in the same cluster at level k, they stay in the same cluster at higher levels
  - _x0001_ dendogram typically shows the similarity of grouped clusters



Hierarchical Clustering: Venn Diagram

- Can also use Venn diagram to show hierarchical clustering, but similarity is not represented quantitatively



Hierarchical Clustering

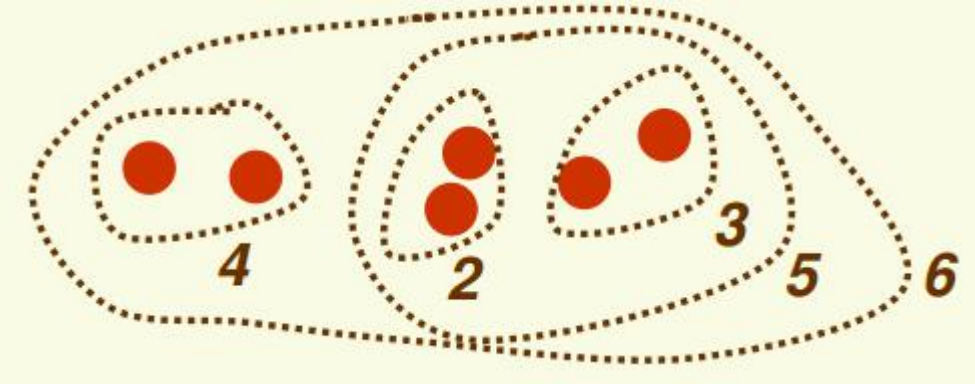
- Algorithms for hierarchical clustering can be divided into two types:
  - 1. Agglomerative (**bottom up**) procedures

    _x0001_ Start with n singleton clusters

    _x0001_ Form hierarchy by merging most similar clusters
  - 2. Divisive (**top bottom**) procedures

    _x0001_ Start with all samples in one cluster

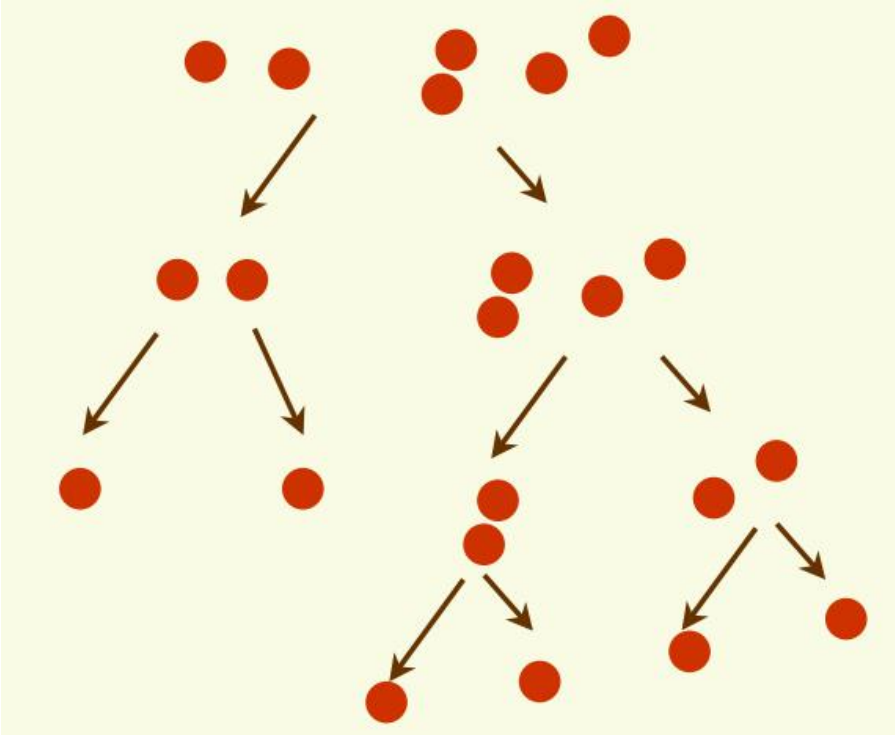
    _x0001_ Form hierarchy by splitting the “worst” clusters



Divisive Hierarchical Clustering

- Any “flat” algorithm which produces a fixed number of clusters can be used

set c = 2

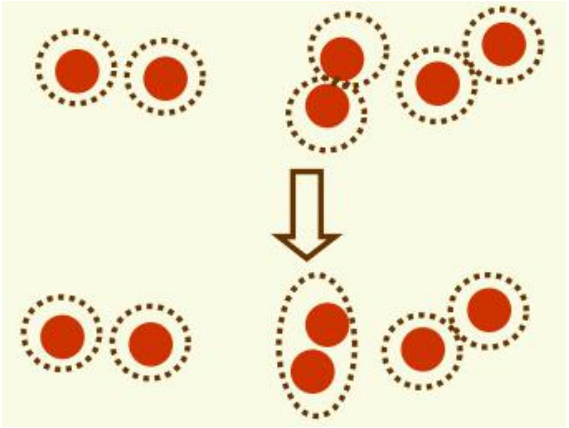


Agglomerative Hierarchical Clustering

- Initialize with each example in singleton cluster while there is more than 1 cluster
  - 1. find 2 nearest clusters
  - 2. merge them

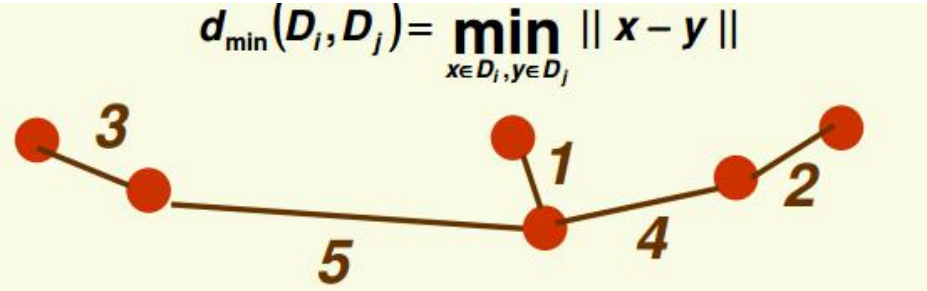
- Four common ways to measure cluster distance

minimum distance	$d_{\min}(D_i, D_j) = \min_{x \in D_i, y \in D_j} \| x - y \|$
maximum distance	$d_{\max}(D_i, D_j) = \max_{x \in D_i, y \in D_j} \| x - y \|$
average distance	$d_{avg}(D_i, D_j) = \frac{1}{n_i n_j} \sum_{x \in D_i} \sum_{y \in D_j} \| x - y \|$
mean distance	$d_{mean}(D_i, D_j) = \| \mu_i - \mu_j \|$

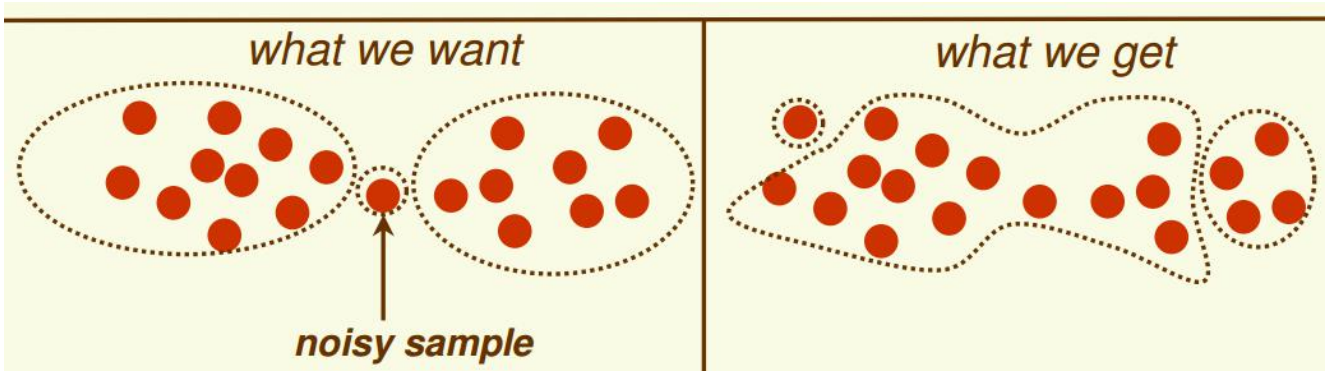


Single Linkage or Nearest Neighbor

- Agglomerative clustering with minimum distance



- generates minimum spanning tree
- encourages growth of elongated clusters
- disadvantage: very sensitive to noise

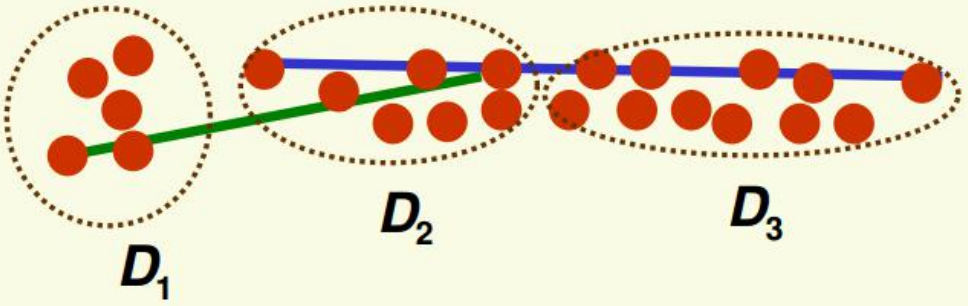


Complete Linkage or Farthest Neighbor

- Agglomerative clustering with maximum distance

$$d_{\max}(D_i, D_j) = \max_{x \in D_i, y \in D_j} \| x - y \|$$

- Encourages compact clusters
- Does not work well if elongated clusters present



- $d_{\max}(D_1, D_2) < d_{\max}(D_2, D_3)$
- thus D_1 and D_2 are merged instead of D_2 and D_3

Average and Mean Agglomerative Clustering

- Agglomerative clustering is more robust under the average or the mean cluster distance

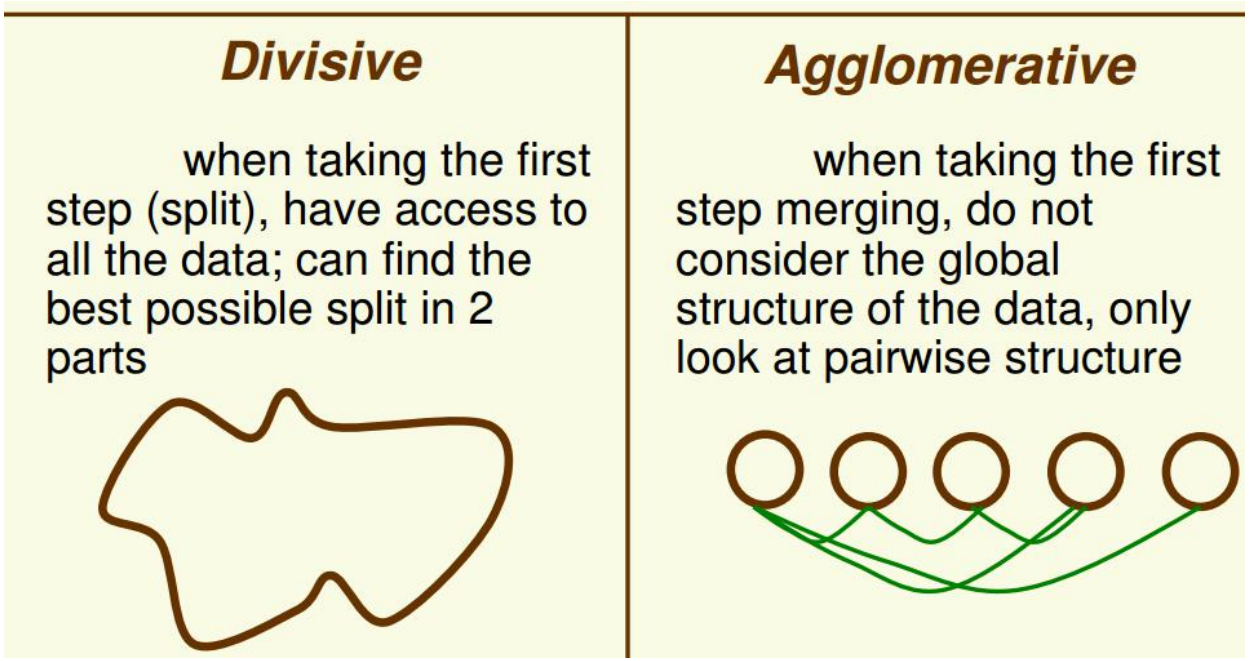
$$d_{avg}(D_i, D_j) = \frac{1}{n_i n_j} \sum_{x \in D_i} \sum_{y \in D_j} || x - y ||$$

$$d_{mean}(D_i, D_j) = || \mu_i - \mu_j ||$$

- Mean distance is cheaper to compute than the average distance
- Unfortunately, there is not much to say about agglomerative clustering theoretically, but it does work reasonably well in practice

Agglomerative vs. Divisive

- Agglomerative is faster to compute, in general
- Divisive may be less "blind" to the global structure of the data

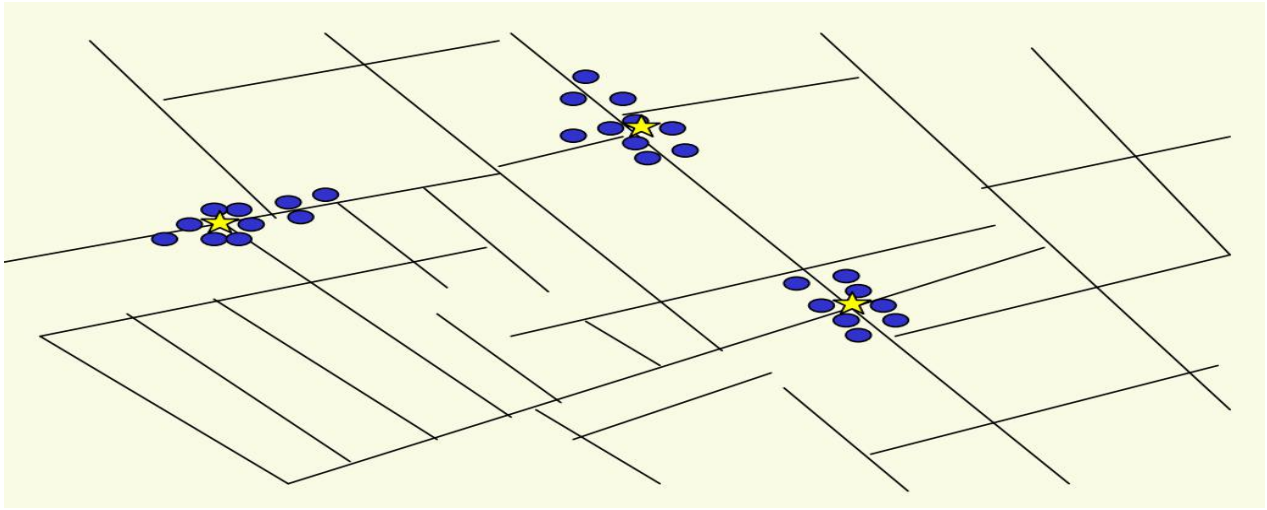


Outline

- Introduce Unsupervised Learning and Clustering
- K-means Algorithm
- Hierarchy Clustering
- **Applications**

Application of Clustering

- John Snow, a London physician plotted the location of cholera deaths on a map during an outbreak in the 1850s.
- The locations indicated that cases were clustered around certain intersections where there were polluted wells -- thus exposing both the problem and the solution.



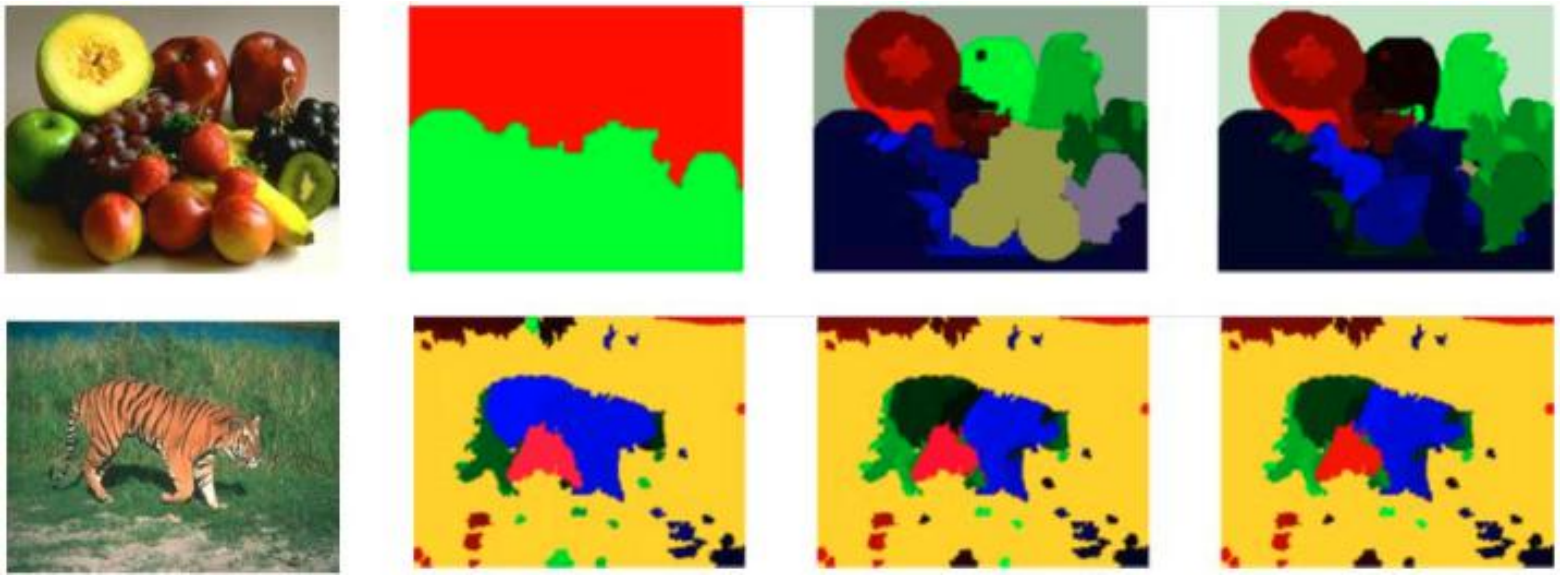
From: Nina Mishra HP Labs

Application of Clustering

- Astronomy
  - SkyCat: Clustered $2x109$ sky objects into stars, galaxies, quasars, etc based on radiation emitted in different spectrum bands.

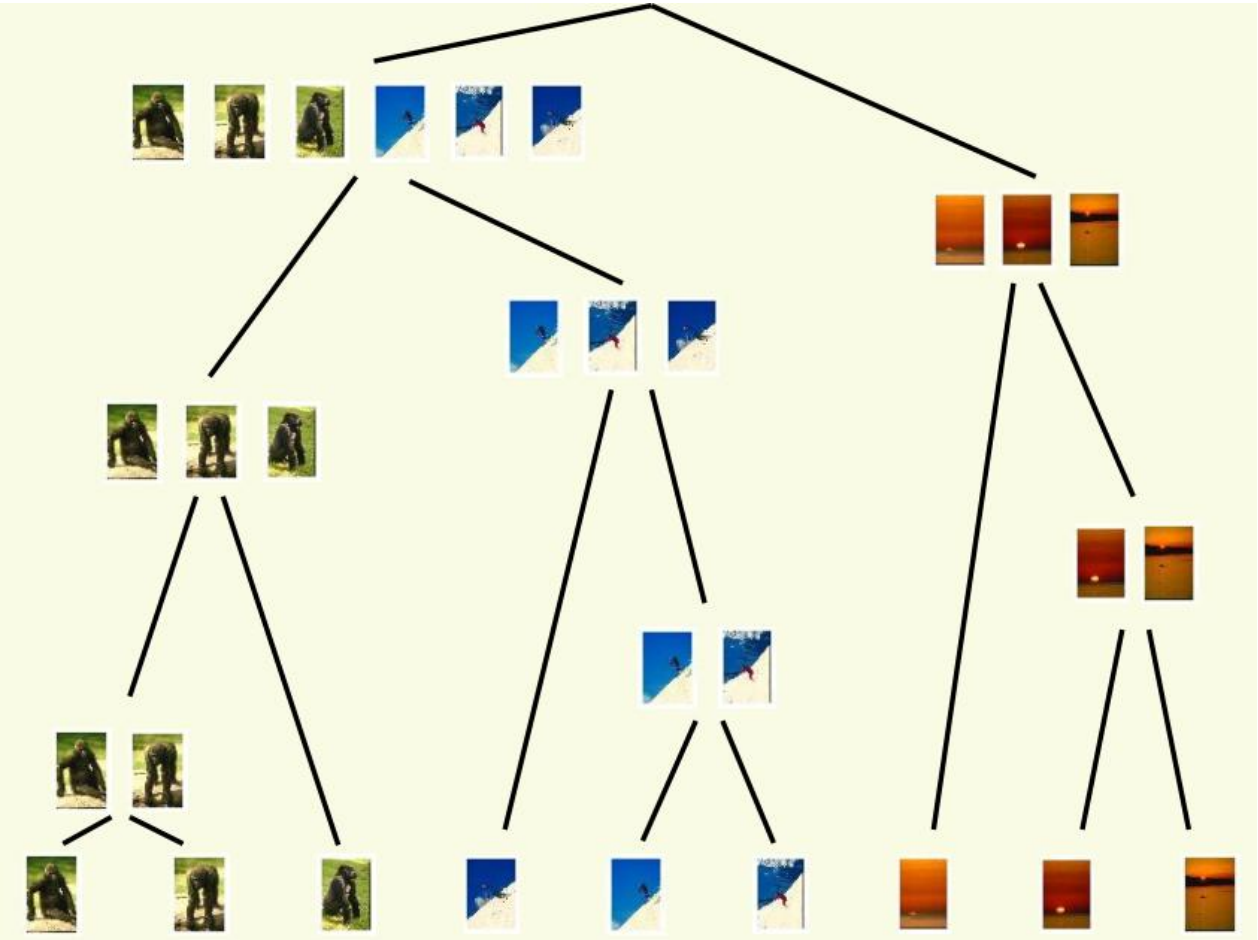
Applications of Clustering

- Image segmentation
  - Find interesting “objects” in images to focus attention at



Applications of Clustering

- Image Database Organization for efficient search



Applications of Clustering

- Data Mining
  - Technology watch

    _x0001_ Derwent Database, contains all patents filed in the last 10 years worldwide

    _x0001_ Searching by keywords leads to thousands of documents

    _x0001_ Find clusters in the database and find if there are any emerging technologies and what competition is up to
  - Marketing

    _x0001_ Customer database

    _x0001_ Find clusters of customers and tailor marketing schemes to them

Applications of Clustering

- Profiling Web Users
  - Use web access logs to generate a feature vector for each user
  - Cluster users based on their feature vectors
  - Identify common goals for users

    _x0001_ Shopping

    _x0001_ Job Seekers

    _x0001_ Product Seekers

    _x0001_ Tutorials Seekers
  - Can use clustering results to improving web content and design

Q & A