CS 461
Artificial Intelligence

Dr. Hashim Yasin

Unsupervised Learning

Unsupervised Learning

- In **unsupervised learning,** the agent learns patterns in the input even though **no explicit feedback** is supplied.
- **Unsupervised learning** occurs when no classifications are given and the *learner must discover categories and regularities in the data*.
- The most general example of unsupervised learning task is **clustering**:
  - potentially useful clusters developed from the input examples.
- For example, **a taxi agent** might gradually develop a concept of "good traffic days" and "bad traffic days".

Clustering

K-means Clustering

- K-means is a partitioning clustering algorithm
- Let the set of data points (or instances) D be

$$\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\},$$

  where
  - $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ir})$ is a vector in a real-valued space $X \subseteq R^r$, and
  - r is the number of attributes (dimensions) in the data.

- The k-means algorithm partitions the given data into k clusters.
  - Each cluster has a cluster **center**, called **centroid**.
  - k is specified by the user

K-means Clustering

- Basic Algorithm:

```
Select K points as the initial centroids.
repeat
    Form K clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change
```

Stopping/Convergence Criterion

1. No (or minimum) re-assignments of data points to different clusters,
2. No (or minimum) change of centroids, or
3. Minimum decrease in the **sum of squared error** (SSE),

$$SSE = \sum_{j=1}^{k} \sum_{\mathbf{x} \in C_j} dist(\mathbf{x}, \mathbf{m}_j)^2$$

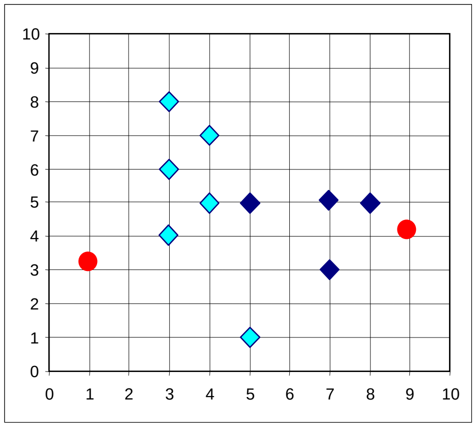
- C_j is the j^{th} cluster, $\mathbf{m}_j$ is the centroid of cluster C_j (the mean vector of all the data points in C_j), and $dist(\mathbf{x}, \mathbf{m}_j)$ is the distance between data point $\mathbf{x}$ and centroid $\mathbf{m}_j$.

K-means Clustering--- Details

- *Initial centroids are often chosen randomly*.
  - Clusters produced vary from one run to another.
- The **centroid** is (typically) the mean of the points in the cluster.
- **‘Closeness’** is measured by Euclidean distance, cosine similarity, correlation, etc.
- K-means will converge for common similarity measures mentioned above.

- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to ‘Until relatively few points change clusters’

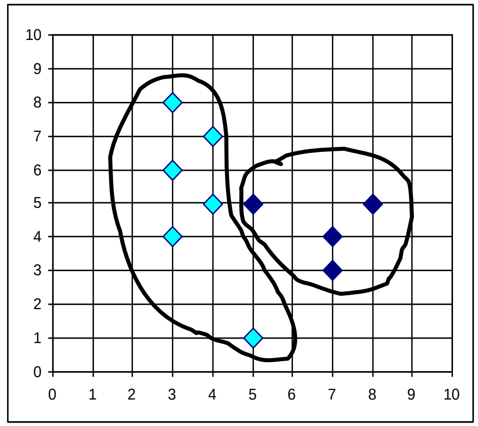
K-means Clustering Example



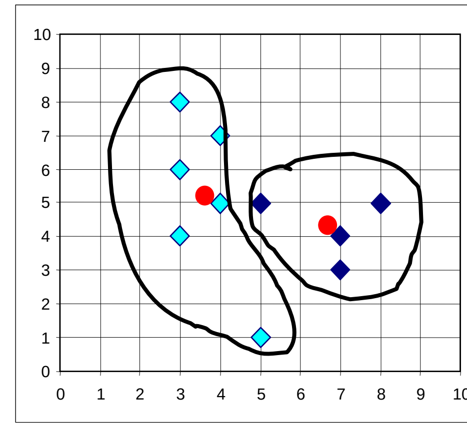
K=2

Arbitrarily choose K object as initial cluster center

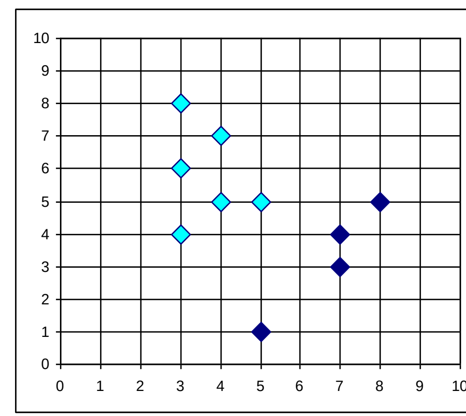
Assign each objects to most similar center



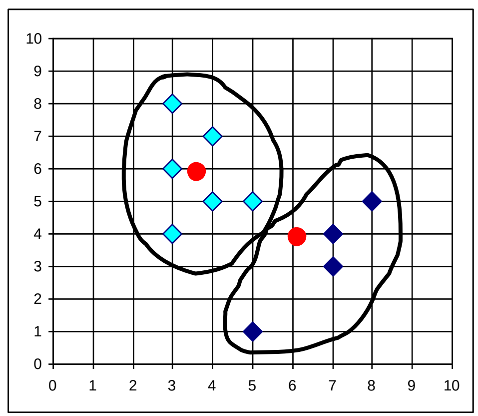
Update the cluster means



reassign



Update the cluster means

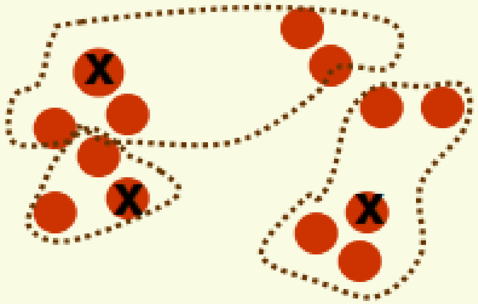


reassign

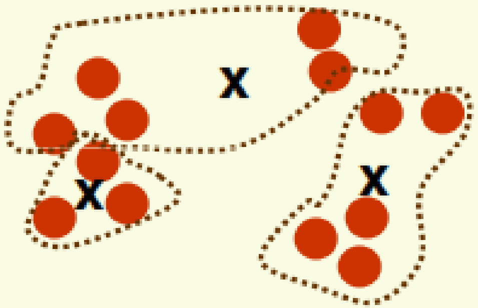
K-means Clustering

k = 3

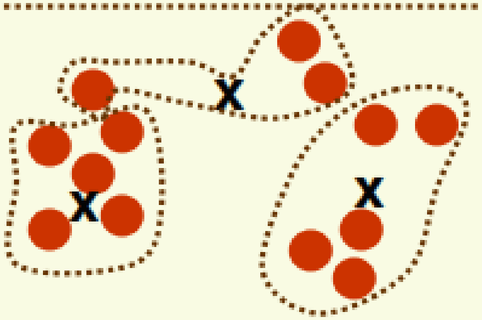
1. Initialize
   - pick ***k*** cluster centers arbitrary
   - assign each example to closest center



2. compute sample means for each cluster



3. reassign all samples to the closest mean



4. if clusters changed at step 3, go to step 2

K-means Clustering

- Pre-processing
  - Normalize the data
  - Eliminate outliers
- Post-processing
  - **Eliminate small clusters** that may represent outliers
  - **Split 'loose' clusters**, i.e., clusters with relatively high SSE
  - **Merge clusters that are 'close'** and that have relatively low SSE

Distance Function

- Most commonly used functions are
  - Euclidean distance and
  - Manhattan (city block) distance
- We denote distance with: *dist*($\mathbf{x}_i$, $\mathbf{x}_j$), where $\mathbf{x}_i$ and $\mathbf{x}_j$ are data points (vectors)
- They are special cases of Minkowski distance. q is positive integer.

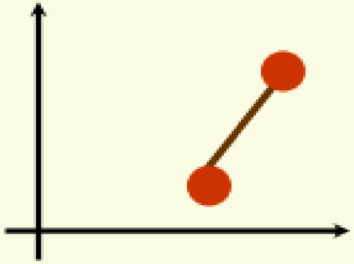
$$d(i,j) = \sqrt[q]{\underbrace{|x_{i1} - x_{j1}|}_{1^{st}\ \text{dimension}}^q + \underbrace{|x_{i2} - x_{j2}|}_{2^{nd}\ \text{dimension}}^q + \ldots + \underbrace{|x_{ip} - x_{jp}|}_{p^{th}\ \text{dimension}}^q}$$

Distance (dissimilarity) Measures

Euclidean distance

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{d} \left(x_i^{(k)} - x_j^{(k)}\right)^2}$$

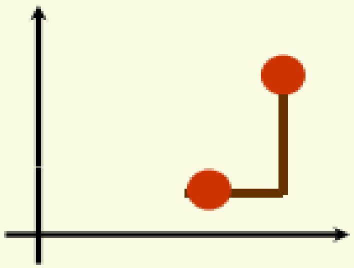
- translation invariant



Manhattan (city block) distance

$$d(x_i, x_j) = \sum_{k=1}^{d} \left| x_i^{(k)} - x_j^{(k)} \right|$$

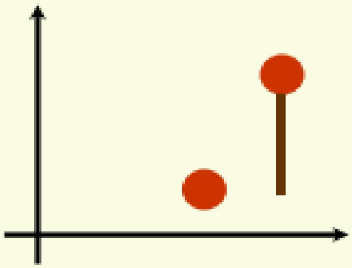
- approximation to Euclidean distance, cheaper to compute



Chebyshev distance

$$d(x_i, x_j) = \max_{1 \le k \le d} | x_i^{(k)} - x_j^{(k)} |$$

- approximation to Euclidean distance, cheapest to compute



K-means Clustering

- Time complexity for K-means clustering is

$$O(\, n \times K \times I \times d \,)$$

  - n = number of points,
  - K = number of clusters,
  - I = number of iterations,
  - d = number of attributes
- The storage required is

$$O((n + K)d)$$

  - n = number of points,
  - K = number of clusters,
  - d = number of attributes

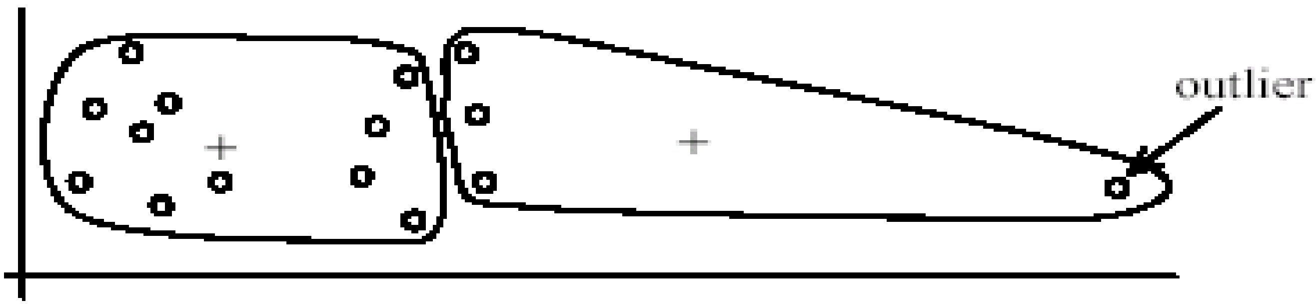
Limitations in K-means Clustering

Limitations of K-means

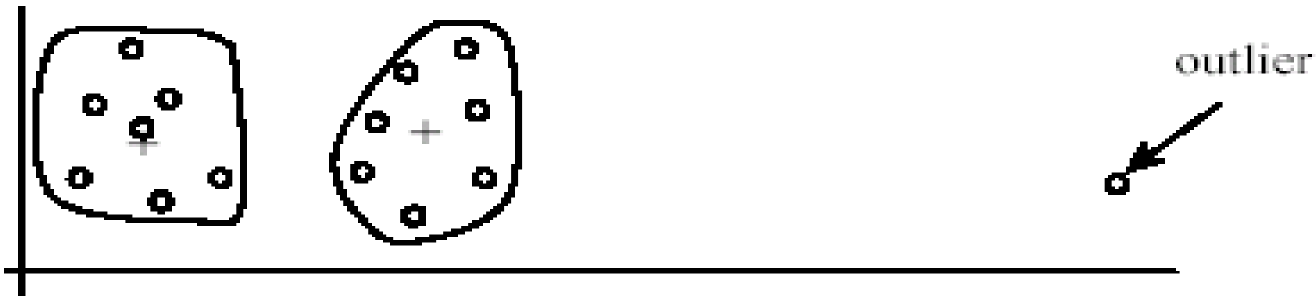
- K-means has problems when the data contains outliers
- *The K-means algorithm is very sensitive to the* ***initial seeds****.*
- K-means has problems when clusters are of different
  - Sizes
  - Densities
  - Non-globular shapes

Limitations of K-means

- K-means has problems when the data contains outliers



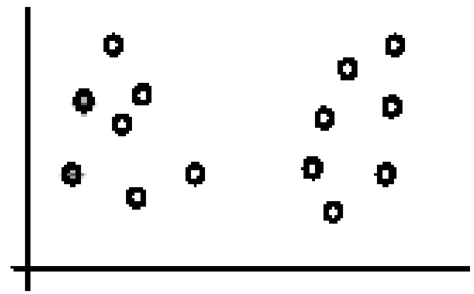
(A): Undesirable clusters



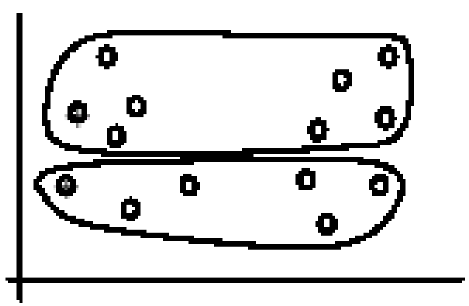
(B): Ideal clusters

Limitations of K-means

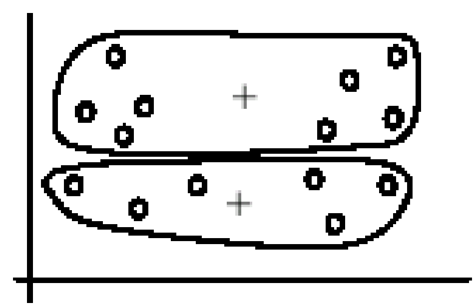
- The algorithm is sensitive to initial seeds



(A). Random selection of seeds (centroids)



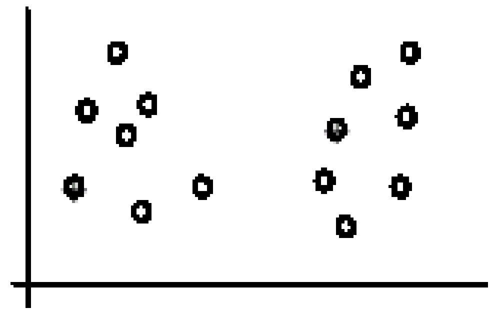
(B). Iteration 1



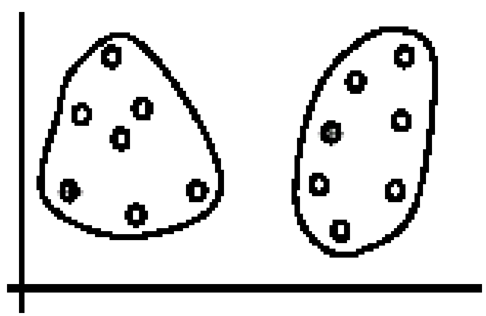
(C). Iteration 2

Limitations of K-means

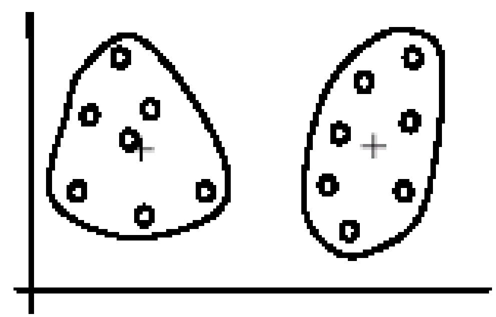
- The algorithm is sensitive to initial seeds



(A). Random selection of k seeds (centroids)



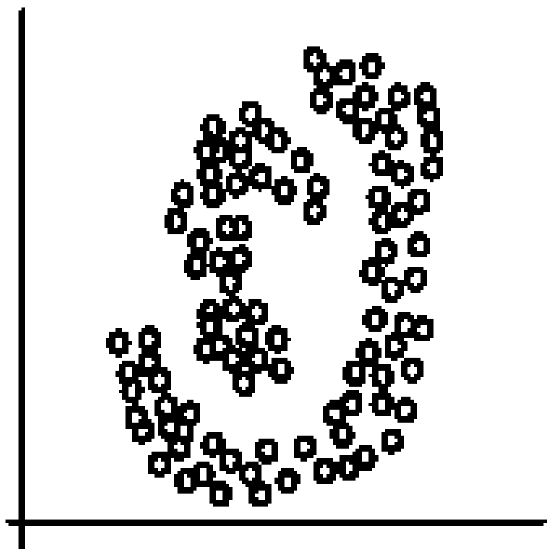
(B). Iteration 1



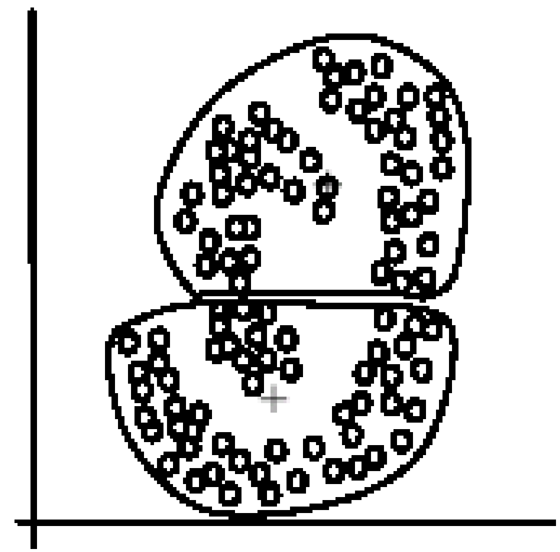
(C). Iteration 2

Limitations of K-means

- 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