Machine Learning

Clustering

Kien C Nguyen

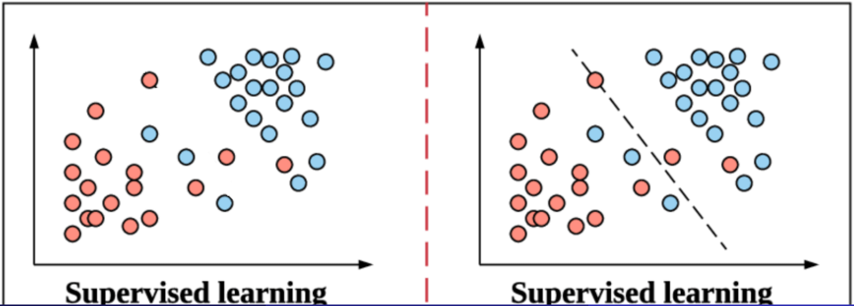
September 4, 2024

Supervised Learning

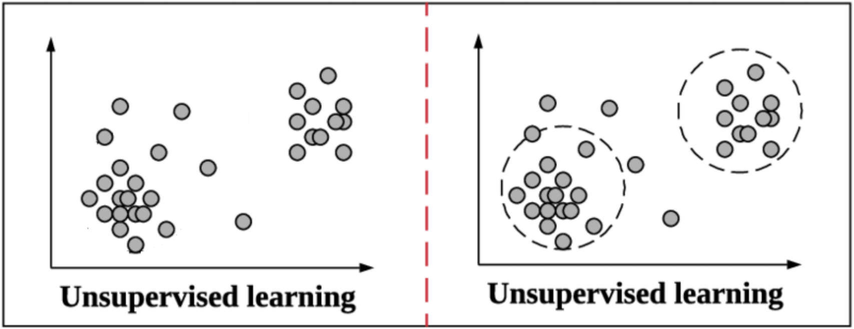
Recall from previous lectures

1. Supervised learning
   - Input : data X and label Y
   - Goal : find a model that minimizes the loss function
2. Why supervised learning?
   - predict outcomes from previous experiences

Figure: (Source: Orchestrating Development Lifecycle of Machine Learning Based IoT Applications: A Survey, Zhenyu Wen)



1. Unsupervised learning
   - Input : data X
   - Goal : group data by finding some common patterns in the features
2. Why unsupervised learning?
   - find features which can be useful for categorization
   - find all kind of unknown patterns in data



Applications of clustering

- **Customer Segmentation (Marketing)**: Group customers based on behavior, demographics, or preferences for targeted marketing.
- **Document Clustering (Text Mining)**: Organize large text datasets, like grouping news articles or research papers by topic.
- **Image Segmentation (Computer Vision)**: Partition images into distinct regions for object detection or medical imaging.
- **Anomaly Detection (Fraud Detection)**: Identify outliers, such as detecting unusual financial transactions or security breaches.
- **Biological Data Analysis (Bioinformatics)**: Group genes or proteins with similar functions or expression patterns.
- **Recommender Systems**: Cluster users or items to provide personalized recommendations in e-commerce or streaming platforms.
- **Social Network Analysis**: Identify communities or subgroups within social networks.
- **Market Basket Analysis (Retail)**: Group products that are often purchased together for cross-selling and store layout optimization.

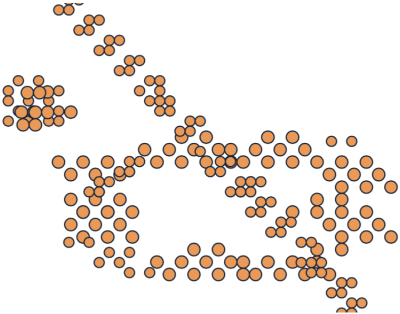
Introduction - Clustering

- Input : $S = \{x^{(i)}\}_{i=1}^{N}$ (N: number of samples), each sample (data point) is a D-dimensional vector

$$x^i = (x_1^i, x_2^i, \ldots, x_D^i)^T$$

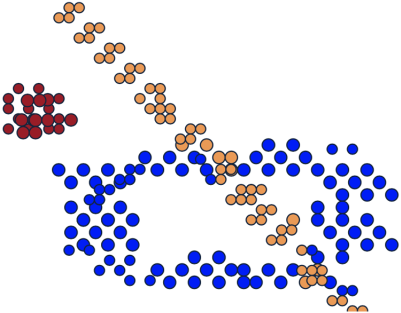
- Output: find structure in the data and organize them into groups.

Figure: Input samples. (Source: UIUC CS446 Lecture notes [1])



- A cluster is a set of samples that are alike
- Samples in different clusters are not alike

Figure: Clustered input samples. (Source: UIUC CS446 Lecture notes [1])



Distance Measures

- A distance measure (metric) is a function $d : R^D \times R^D \to R$ that satisfies
  1. $d(\mathbf{x}, \mathbf{y}) \geq 0, \quad d(\mathbf{x}, \mathbf{y}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{y}$
  2. $d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z}) \geq d(\mathbf{x}, \mathbf{z})$ (Triangle inequality)
  3. $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$ (Symmetry)
- For the purpose of clustering, sometimes we can use distances that are not a metric (e.g. those that do not satisfy triangle inequality or symmetry.)

Distance Measures

- L^2 distance (Euclidean distance)

$$\begin{aligned} d(\mathbf{x}, \mathbf{y}) &= \|(\mathbf{x}-\mathbf{y})\|_2 = \sqrt{(\mathbf{x}-\mathbf{y})^2} \\ &= \sqrt{(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y})} = \sqrt{\sum_{i=1}^{D}(x_i - y_i)^2} \end{aligned}$$

- L^1 distance (Manhattan distance)

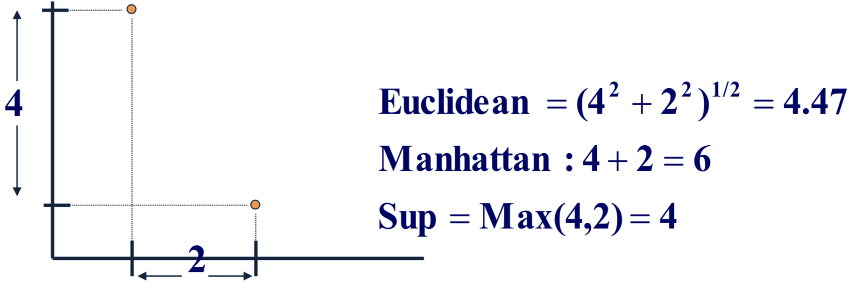
$$d(\mathbf{x}, \mathbf{y}) = \|(\mathbf{x}-\mathbf{y})\|_1 = \sum_{i=1}^{D}|x_i - y_i|$$

- L^∞ distance (sup distance)

$$d(\mathbf{x}, \mathbf{y}) = \|(\mathbf{x}-\mathbf{y})\|_\infty = max_{1 \le i \le D}|x_i - y_i|$$

Distance measures

Figure: Different types of distance measures. (Source: UIUC CS446 Lecture notes [1])



Distance measures

- We are given a matrix of distances between any pair of samples.

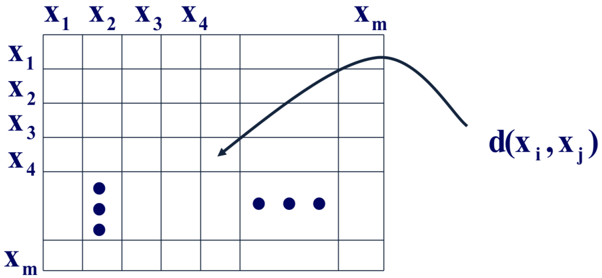
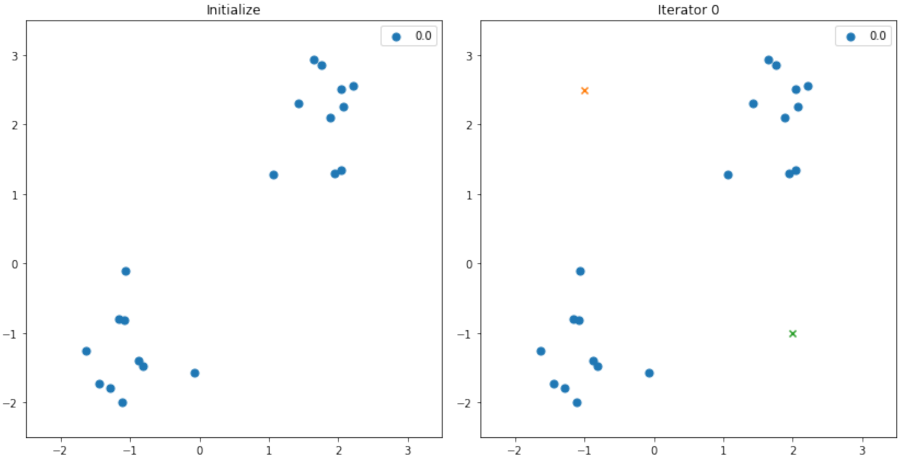
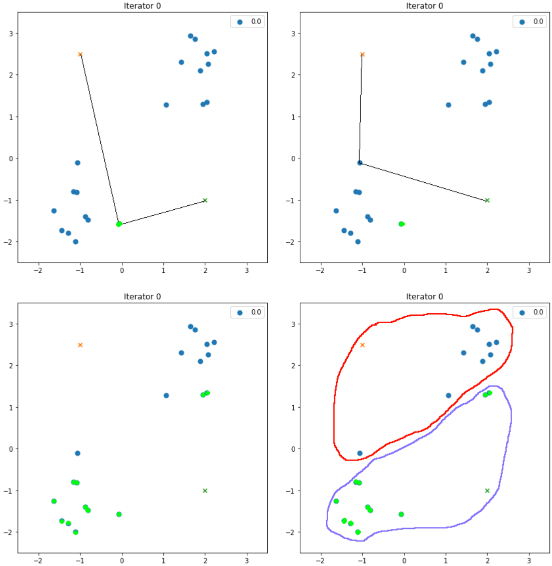


Figure: Matrix of distances. (Source: UIUC CS446 Lecture notes [1])

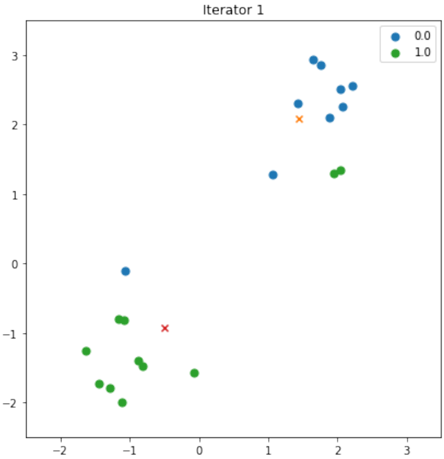
K-means Algorithm



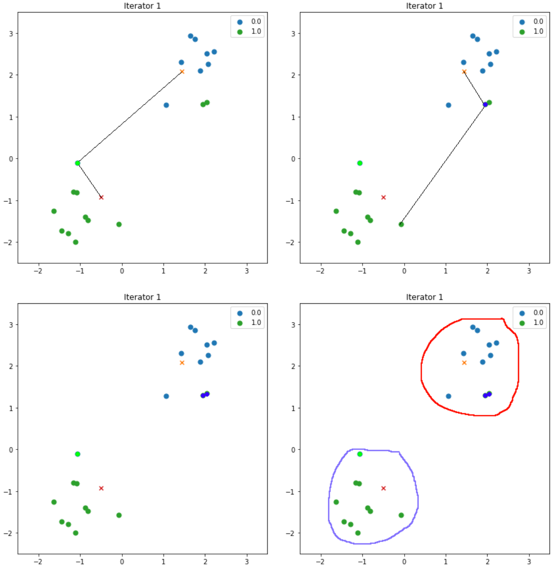
K-means Algorithm

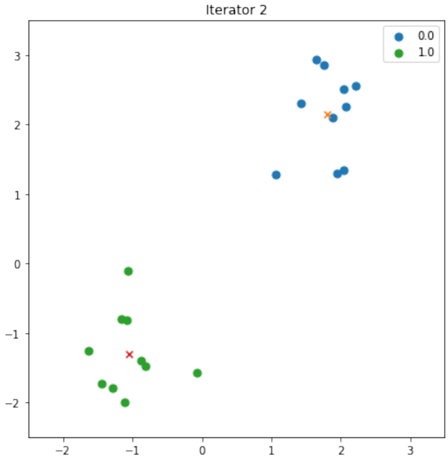


K-means Algorithm



K-means Algorithm




Iterator 2
0.0
1.0
3
2
1
0
−1
−2
−2
−1
0
1
2
3

K-means Algorithm

Input:

- K (number of clusters)
- $\{x^{(i)}\}_{i=1}^{N}$

Initialization:

Randomly initialize K cluster centroids $\mu_1, \mu_2, \ldots, \mu_K \in \mathbb{R}^D$

while *Assignment changes from the last iteration* **do**

  Assignment:

  for $i = 1$ *to* N **do**

    Assign $x^{(i)}$ to the cluster with the minimum distance $d(x^{(i)}, \mu_k)$

  end

  Update:

  for j=*1 to* K **do**

    $\mu_k =$ mean of all the points assigned to cluster k

  end

end

Algorithm 1: K-means Algorithm

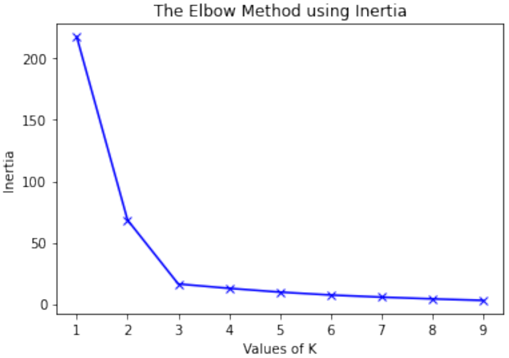
Challenges of K-means

- Different K different outputs.
- With same K, the output won't be always the same because of the randomly initial centroids.
- Due to the nature of Euclidean distance, it is not a suitable algorithm when dealing with clusters that adopt non-spherical shapes.

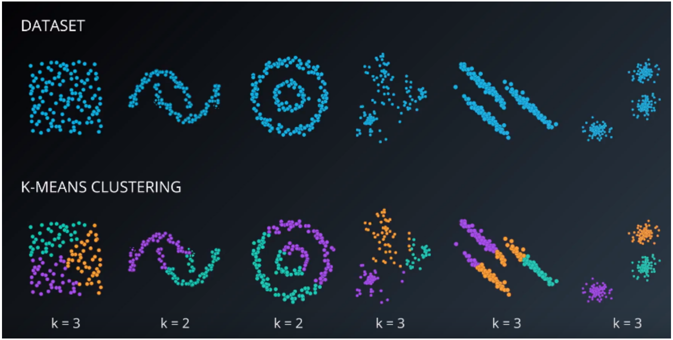
- Field knowledge
- Business decision
- Elbow Method

Elbow Method

- The elbow method is used for determining the correct number of clusters in a dataset.
- How it works? Plot the cost function against K and choose K using the "elbow" method.



- K-means clustering with spherical-shaped distributions



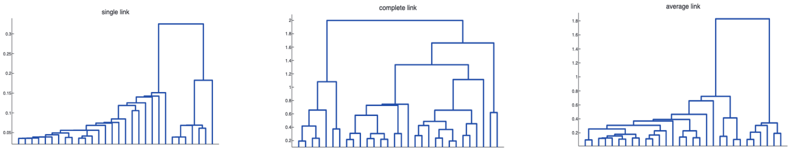
Main approaches:

- Bottom-up/Agglomerative clustering: each data point starts in its own cluster
- Top-down/Divisive clustering: all data points start in the same cluster

Input:
$\{x^{(i)}\}_{i=1}^{N}$
Initialization:
Clusters as singletons C_i for $i \in \{1, ..., N\}$ and set of clusters available for merging $S \leftarrow \{1, ..., n\}$
while *There are available clusters for merging* **do**
    Pick 2 most similar clusters to merge: $(j, k) \leftarrow_{j,k \in S} d_{j,k}$
    Create new cluster $C_{lj} \cup C_k$
    Mark j and k as unavailable: $S \leftarrow S \setminus \{j, k\}$
    if $C_l \neq \{1, ..., N\}$ **then**
        Mark l as available, $S \leftarrow S \cup \{l\}$
    end
    Update:
    for $i \in S$ **do**
        Update dissimilarity matrix $d(i, l)$
    end
end

Algorithm 2: Agglomerative clustering

Different variants of agglomerative clustering:



(a) Single linkage (b) Complete linkage (c) Average linkage

Figure: Hierarchical clustering of yeast gene expression data

Single link	Complete link	Average link
$min_{i \in G, j \in H} d_{i,j}$	$max_{i \in G, j \in H} d_{i,j}$	$\frac{1}{N_G N_H} \sum_{i \in G} \sum_{j \in H} d_{i,j}$

Table: Distance between two clusters $d(G, H)$

- NO need of defining k - number of clusters
- Easy to implement and dendrogram produced is very useful in understanding the data
- Time complexity $O(nlogn)$ (compare with k-Mean)
- Sensitivity to noise and outliers, breaking large clusters, difficulty handling different sized clusters and convex shapes
- NO backtracking, NO object function

Other Unsupervised Learning Models

- Density-Based Spatial Clustering of Applications with Noise (DBSCAN)
- Gaussian Mixture Models (GMM)
- Principal component analysis (PCA)

References

[1] UIUC CS 446 Machine Learning
[2] Andrew Ng – Coursera Machine Learning
[3] https://towardsdatascience.com/unsupervised-machine-learning-clustering-analysis-d40f2b34ae7e
[4] K. P. Murpy – Machine Learning – A Probabilistic Perspective, MIT Press, 2012