K-Means Clustering

Agenda

1. Discussion Questions

2. Unsupervised Learning and Clustering

3. Common Distance Metrics

4. Scaling

5. K-Means Clustering

6. Optimal number of clusters

7. Pros & cons

Questions to discuss

1.   What is Unsupervised Learning and Clustering?

2.   What is K-Means Clustering and how it works?

3.   How to find the optimal number of clusters?

4.   How to use t-SNE to visualize clusters?

Unsupervised Learning

- Unsupervised Learning is a class of Machine Learning techniques to find the patterns in data.

- The data given to unsupervised algorithms are not labeled, which means only the input variables (X) are given with no corresponding output variable (y).

- Involves training of an algorithm using information that is neither classified nor labeled.

- No defined dependent and independent variables.

- Patterns in the data are used to identify/group similar observations

Clustering

- The objective is to group a set of objects in such a way that objects in the same group are more similar to each other than to those in other groups

- It involves ensuring that the distance between data points in a cluster is very low compared to the distance between 2 clusters.

- This kind of algorithm captures the hidden patterns in data to find the underlying structure and discover new insights.

- The similarity between data points is determined by the distance between them, which can be measured using different distance metrics

Scattered points

Clustered points
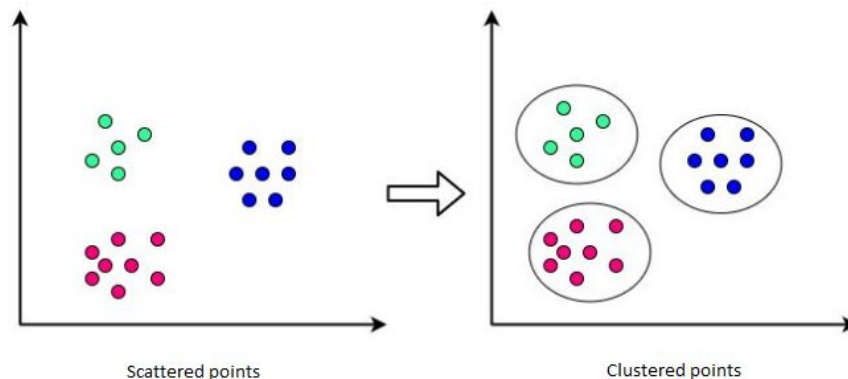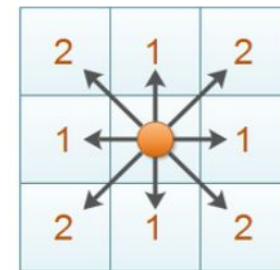
Common Distance Metrics

- **Euclidean Distance Metrics**

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$
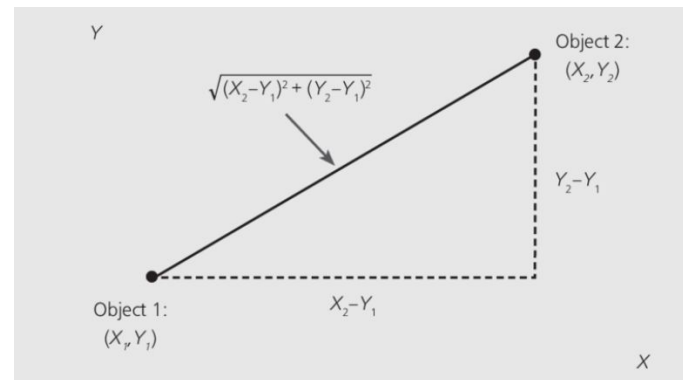
- **Manhattan distances**

$$\sum_{i=1}^{k}|x_i - y_i|$$
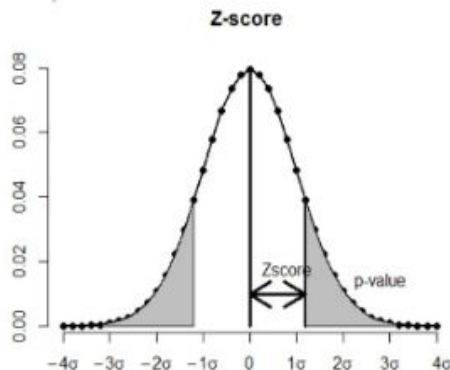
$$|x_1 - x_2| + |y_1 - y_2|$$

Scaling the data

- It is important to normalize the data using either Z-score or StandardScaler before performing K-means clustering

- This ensures that the different attributes in the data are of the same scale


$$Z = \frac{x - \mu}{\sigma}$$

K-Means Clustering

- K-Means is one of the most common clustering techniques

- It is a centroid-based clustering algorithm where the objective is to find K clusters / groups

- The working of K-means clustering can be summarized as follows:

  - Step 1: Initialize the K random centroids or K points
  - Step 2: For each data point, calculate the Euclidean distance of it from randomly chosen K centroids and assign each point to a minimum distance cluster.
  - Step 3: Update the centroid by using newly assigned data points to the cluster by calculating the average of data points.
  - Step 4: Repeat the above process for a given no. of iterations or until the centroid allocation no longer changes

- Large K produces smaller groups and small K produces larger groups

Optimal Number of Clusters: Elbow Method

- There is no method to define the exact value of K

- The Elbow method is the most popular and well-known method to find the optimal no. of clusters

- This method is based on plotting the value of the cost function against different values of K

- The point where the distortion declines most is said to be the elbow point and defines the optimal number of clusters for the dataset
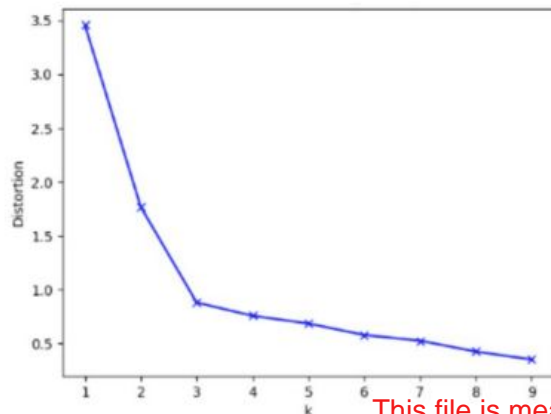
- In the example here, you can that the distortion decreases most at 3

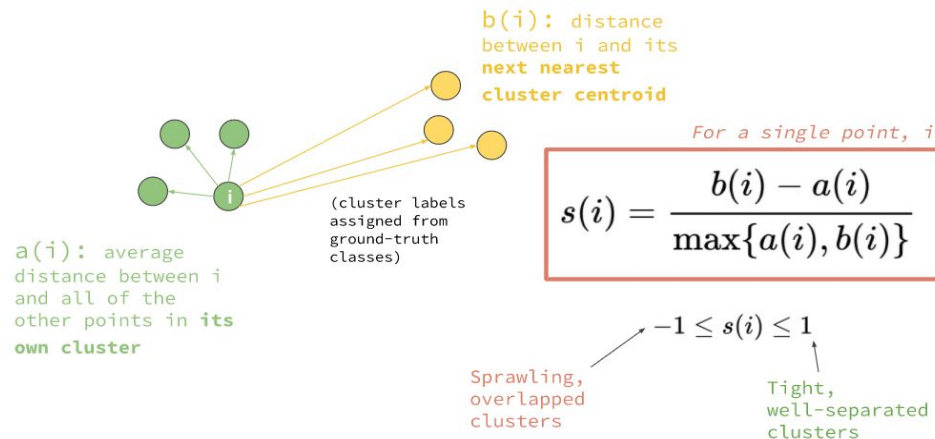- Hence, the optimal value of K will be 3 for performing the clustering

Optimal Number of Clusters: Silhouette Score

- The silhouette score is a metric which indicates the goodness of clustering algorithms

- It values range between -1 to +1

  - 1 indicates tight, well-separated clusters
  - 0 indicates clusters not well separable
  - -1 indicates data points of one cluster is more closer to centroid of another cluster than the centroid of its own cluster

- **Silhouette score = (b-a)/max(a,b)**

  - a = average intra-cluster distance i.e., the average distance between each point within a cluster.
 - b = average inter-cluster distance i.e., the average distance between all clusters.

b(i): distance between i and its **next nearest cluster centroid**

(cluster labels assigned from ground-truth classes)

a(i): average distance between i and all of the other points in **its own cluster**

For a single point, i

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

$$-1 \leq s(i) \leq 1$$

Sprawling, overlapped clusters

Tight, well-separated clusters

t-SNE: Overview

- t-SNE stands for **t**-distributed **S**tochastic **N**eighbor **E**mbedding

- Helps visualize high-dimensional data in 2 or 3 dimensions.

- Preserves local data structure when transforming from higher to lower dimensions
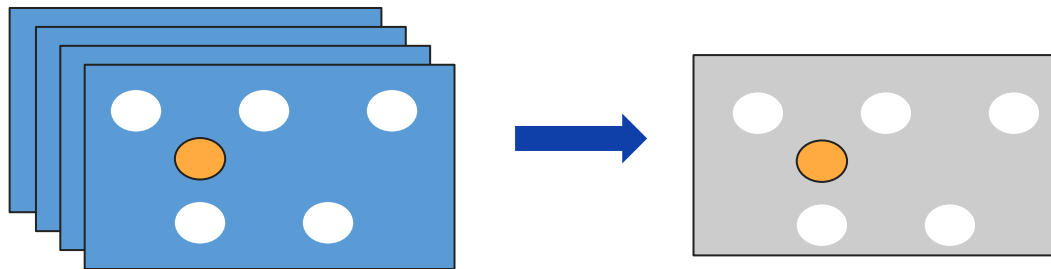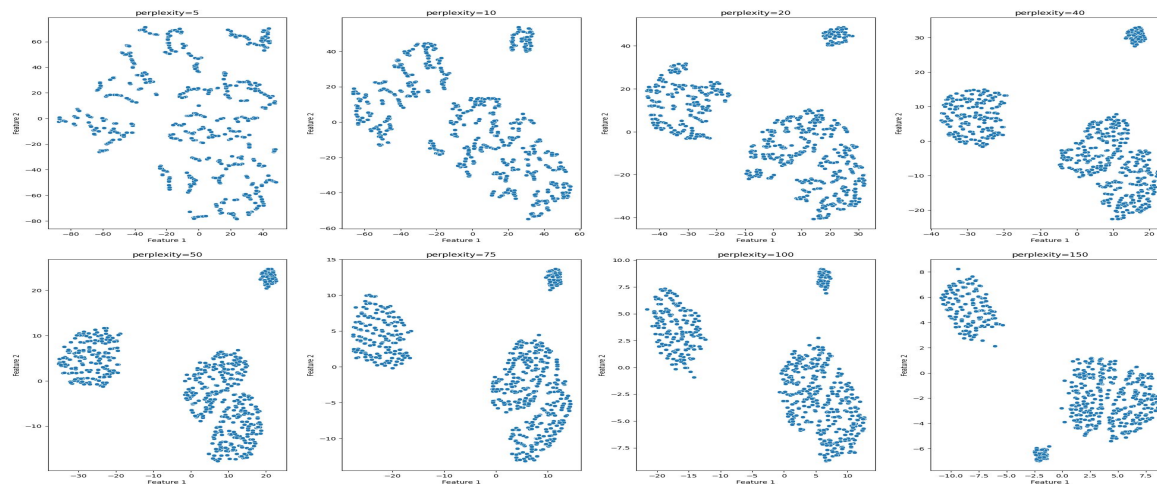
t-SNE: Algorithm

- Initialize with random points in a lower-dimensional space.

- Compute pairwise similarities in the high-dimensional space.

- Compute pairwise similarities in the low-dimensional space.

- Define a loss function that measures the difference between the high-dimensional and low-dimensional similarities.

- Minimize the loss function using optimization techniques (e.g., gradient descent).

- Update the points in the low-dimensional space iteratively.

- Continue until convergence or a set number of iterations.

t-SNE: Perplexity

- A parameter in t-SNE that controls the balance between local and global aspects of data.

- Influences the number of nearest neighbors considered for each data point.

- A low perplexity emphasizes local structure; a high perplexity captures global structure.

- Common range for perplexity values is between 5 and 50.

Perplexity values: 5,10,20,40,50,75,100,150

Happy Learning !