Review Questions

1. Kmeans-why euclidean distance over other methods?

-It’s fast, robust and easier to understand.

- Relatively efficient: O(tknd), where n is number of objects,

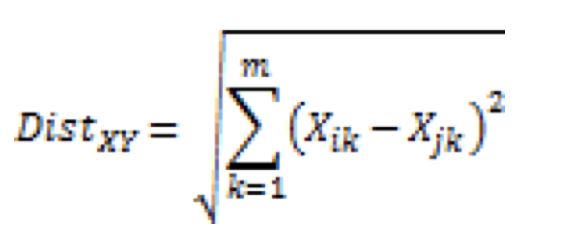
k is number of clusters, d is number of dimension of each

object, and t is number of iterations. Normally, k, t d < n.

-Gives best result when data set are distinct or well

separated from each other.

-Euclidean distance has several desirable properties, such as

being symmetric, positive definite, and satisfying the triangle

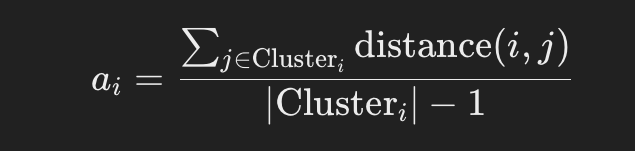
Inequality.

2) How to analyse and explain the math behind silhouette score.

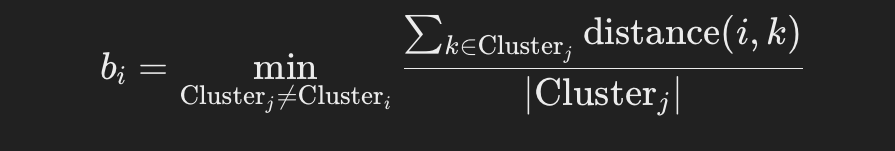
The Silhouette score assesses clustering quality by examining how well a data point aligns with its cluster (cohesion) versus how distinct it is from points in other clusters (separation).

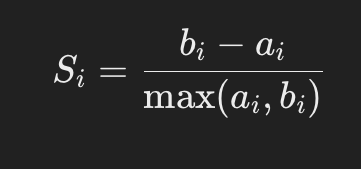
A>Calculate the mean distance between a data point and all

other points in the same cluster. This is the cohesion

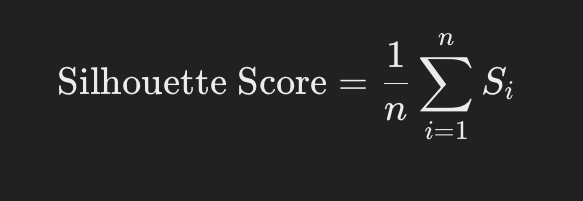
measurement.

B>Calculate the mean distance between a data point and all points in the next nearest cluster. This is the separation

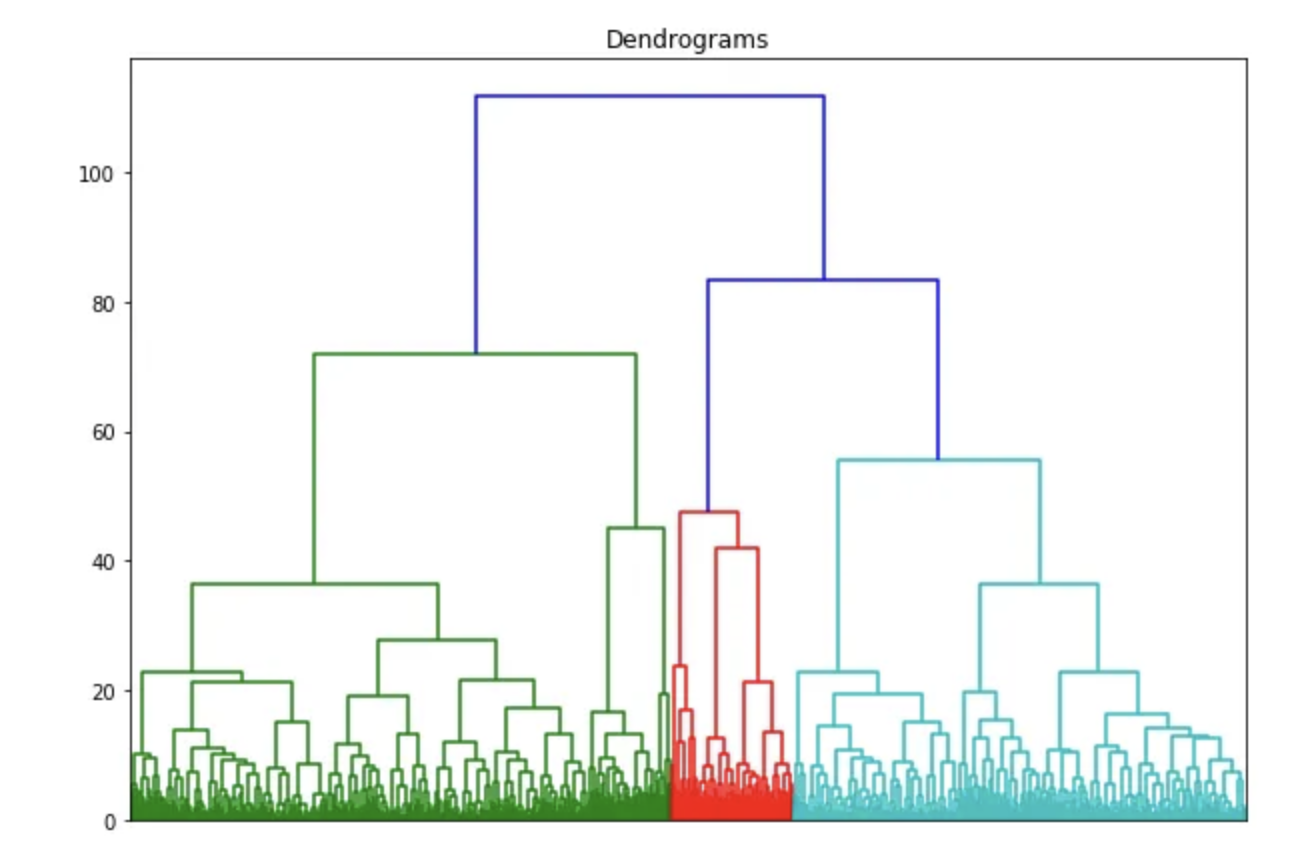
measurement.

the Silhouette score for each data point(i)

The overall Silhouette score for the clustering solution is the

average of the Silhouette scores for all data points.

Here n =total number of datapoints

3) How to choose the number of clusters in

hierarchical clustering ?

To get the optimal number of clusters for hierarchical clustering, we make use a dendrogram which is tree-like chart that shows the sequences of merges or splits of clusters.

If two clusters are merged, the

dendrogram will join them in a graph and the height of the join will be the distance between those clusters.

We plot graph using dendrogram function from script library

import scipy.cluster.hierarchy as shc

4) DBSCAN

Density-Based Spatial Clustering Of Applications With Noise

Clusters represent areas of high density within the dataset, which are surrounded by regions containing fewer data points. The DBSCAN algorithm operates on this concept of identifying clusters and noise. Its fundamental principle is that every point within a cluster must have a minimum number of neighboring points within a specified radius.

In DBSCAN we don’t need to specify number of clusters.

DBSCAN has two key parameters:

1.Epsilon (ε) - A threshold value that determines the maximum radius of the neighborhood around a point. Points within this radius are considered neighbors.

2.MinPts - The minimum number of points required in a neighborhood to form a dense region.

The DBSCAN algorithm works as follows:

1. For each point in the dataset, the algorithm calculates the number of points within the ε radius.
2. If there are at least MinPts points in the ε radius, the algorithm forms a new cluster, including all points in the neighborhood.
3. The algorithm then iteratively adds new neighboring points to the cluster, expanding its radius until no more points can be added.
4. The algorithm repeats steps 1-3 for all remaining unvisited points in the dataset.
5. Points that do not belong to any cluster are marked as outliers or noise.

5) What are landmarks, kernels and similarity

functions in svm?

Landmarks in Kernelized SVM:

Landmarks are a subset of data points used in the kernel trick of SVMs.(The kernel trick is used in SVMs and other models to handle non-linear relationships in data without explicitly transforming it into a higher-dimensional space.)

They reduce computational complexity by approximating dot products in a higher-dimensional space.

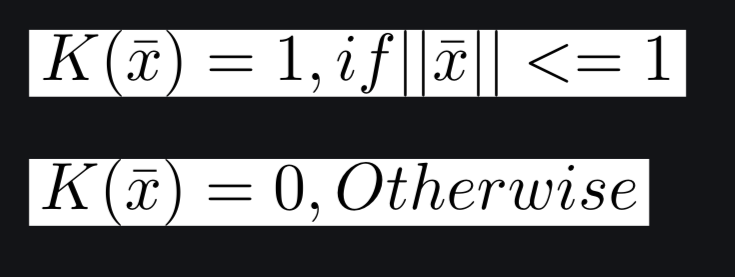
Landmarks are representative samples selected from the training set.

The kernel function transforms input data for processing in SVMs by allowing manipulation through mathematical functions. It

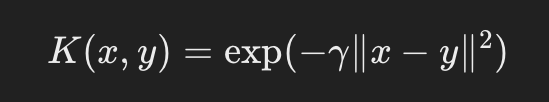
converts the training data to represent non-linear decision boundaries as linear equations in higher-dimensional spaces,

calculating inner products between points in the feature

dimension.

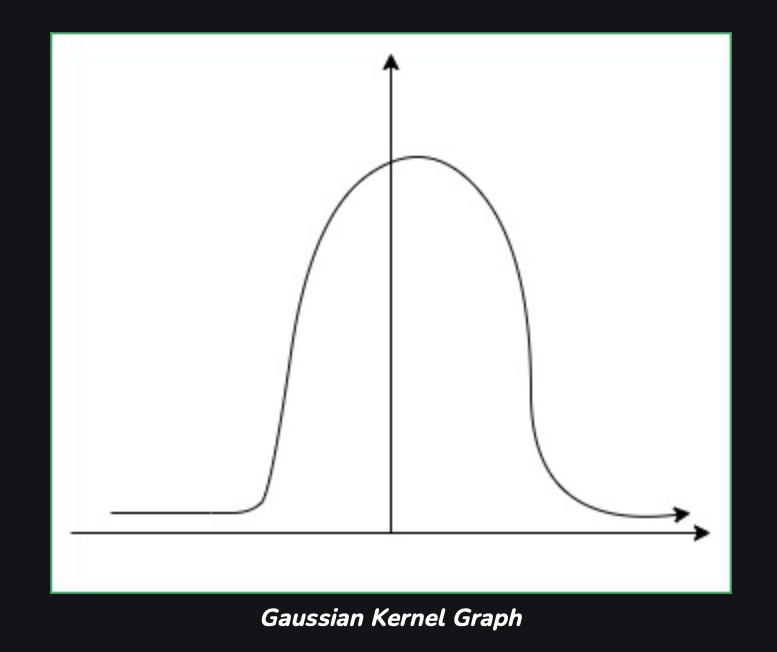
Standard kernel equation-

1.Gaussian Kernel Radial Basis Function (RBF)

It is used to perform transformation when there is no prior knowledge about data by radial basis method.

is the euclidean distance between data points x and y.

= parameter that controls smoothness of graph



Similarity function (widely refereed as the *kernel function)* is a function that is used to assess the similarity between two

data-points. Given two data-points it outputs a similarity score.

The simplest example is of linear kernel, also called dot-product. Given two vectors, the similarity is the length of the projection of one vector on another.

1. Preprocess data:
   * Load "titanic.csv" using a library like pandas.
   * Preprocess missing values using imputation or
   * deletion(NaN).
   * Encode categorical features using techniques like one-hot encoding.
   * Split data into training and testing sets.
2. Define SVM class:
   * Create a Python class SVM with attributes like learning rate, kernel function, and regularization parameter.
3. Kernel functions:
   * Implement methods for different kernel functions (e.g., linear, polynomial, Gaussian).

GITHUB LINK----------https://github.com/rampofin/mrm.git