Image Clustering

Name - Saksham Nayyar

Username – Sakshamnayyar

G no. – G01462522

George mason University

# Introduction

This assignment centers on the implementation of a custom K-Means algorithm for clustering 10,740 handwritten digit images. Each image, initially presented as a 28x28 pixel matrix, is flattened into a 1x784 vector. The goal is to discern patterns within the data and cluster the images accordingly, with no provided labels. This report provides a brief overview of the methodology, focusing on algorithmic intricacies, preprocessing steps, and the chosen evaluation metric. The emphasis lies on the critical role of thoughtful metric selection and preprocessing in the realm of image data clustering.

# Program Instructions

To run the program and reproduce the results presented in the report, follow these instructions:

1. Download file project file.
2. Add the path to dataset file in the code.
3. Execute python file (HW3-G01462522.py), after the execution a new file will get created in the same folder containing the results of the dataset (format.dat).
4. Now you can compare the result in format.dat file to the ground truth to determine accuracy.

# Methodology

In this assignment, I have implemented my own custom K-Means clustering algorithm that uses Lloyd’s algorithm to make clusters of the data. Below I have mentioned all the steps of the algorithm:

1. **Centroid Initializing**: I have used K-Means++ to initialize centroids. Began with the selection of first centroid randomly and then iteratively selected subsequent centroids with probabilities based on squared distances to the nearest existing centroid.
2. **Data point assignment**: Assign each data point to the cluster represented by the minimum distance.
3. **Centroid update**: Update centroids by calculating the mean of assigned data points.
4. **Convergence check**: Keep updating centroids until the maximum distance between old and new centroids is below tolerance or the total number of iterations are completed.

# Data Pre-processing

In this assignment, I have used three techniques to preprocess my data before feeding it to K-Means model.

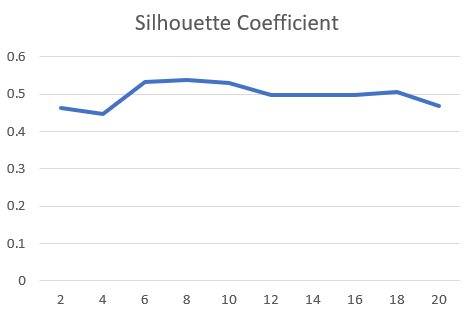
1. **Remove Constant Columns**: Eliminate features with constant values, because the carry no information. It also helps in reducing noise and mitigates potential noise from features.
2. **MinMax Scalling:** Standardize features by transforming values to a specified range i.e.(0,1)
3. **UMAP(Uniform Manifold Approximation and Projection):** This is a dimensionality reduction technique that reduces dimensionality while preserving local and globaldata structures. It enables visualization of complex relation ships within the data. Using RandomisedSearchCV and silhouette Coefficient to determine the best parameters that can be used for UMAP. Best parameters came out to be min\_dist = 0.24868, n\_components = 22 and n\_neighbours = 6. But did not went through with these parameters, as using these parameters v-measure score on miner was coming out to be 0.73 which is lower than if I choose n\_components = 2, that was 0.77.

The processed dataset is expected to exhibit reduced noise, normalized scales and improved representation of intrinsic data structure, fostering more accurate and efficient clustering.

# Evaluation Metric

For internal evaluation, **Silhouette Coefficient** has been adopted as a metric to gauge the quality of the clustering model. The silhouette coefficient measures the compactness of clusters and separation between clusters, offering insights into the consistency and distinctiveness of the identified clusters. With values ranging from -1 to 1, higher value indicates well defined clusters with instances tightly packed within their own clusters and well-separated from neighboring clusters. The Silhouette Coefficient serves as a valuable complement to the external metric (V-measure) used for leaderboard ranking, offering a more nuanced perspective on the clustering performance.

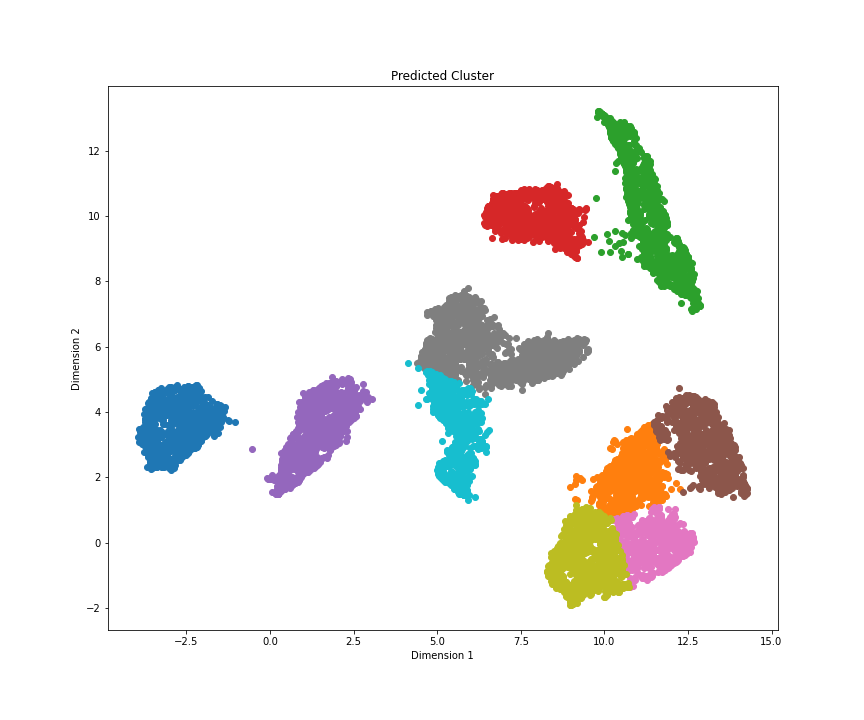
To assess clustering quality across varying values of K (number of clusters). Silhouette Coefficient has been computed for values of K ranging from 2 to 20 in steps of 2. If I examine these values, the optimal value of k comes up to be 4.



Silhouette Coefficient vs K (number of clusters).

# Results

* **Performance Metrics:** My final model achieved a V-measure score of 0.77 on the test dataset as shown on the miner and Silhouette Coefficient of 0.529.
* **Miner Submission:** Miner Rank for the latest submissionis rank 48. That includes multiple submissions from one person. If we don’t consider that, rank is 11.

****

A graph showing different colored clusters formed after performing K-Means Clustering algorithm.