Programming Assignment 3
Mining Massive Data
University of Vienna

Group 08
Student: Zoheir El Houari
Matriculation: a12044027

Remark:

This assignment was done only by Zoheir El Houari, I tried to contact my teammates but my attempts end up with failure, I'm assuming they dropped the class.

Task 1: Lloyd’s algorithm for k-Means Clustering

Goal:

Cluster the dataset into k different clusters. Each sample is assigned to the cluster with the nearest mean value

- The K-Means algorithm was implemented following Lloyd's algorithm steps
- 2 distance functions we tested to see their impact on clustering accuracy NMI

Implementation:

K-means algorithm uses an iterative refinement techniques, the algorithm alternates between the following steps:

- **Assignment step:** Once a set of k centroids is available (first iteration it is assigned to first k points), the clusters are reassigned to points to contain the points closest in distance to each centroid.

- **Update step:** Given a set of clusters, the centroids values are recalculated as the mean values of all points belonging to a cluster.

The two-steps are repeated until the assignments of clusters and centroids no longer change, meaning that the distance between ***old_centroids*** and ***self.centroids*** is zero. This is when the algorithm converges and stops the execution.

Results:

- Number of iterations needed for convergence:

Since the algorithm took ***4882.666*** seconds (around 82min) for 25 iterations and didn't converge, it was very computationally expensive to run other tests and increase the number of iterations to see if the algorithm would converge eventually.

- Runtime for the algorithm over 5 Iterations: **1222.9526546001434** seconds
- Achieved NMI over 5 iterations:

I experimented with the following distance functions and I recalculated the NMI for both of them

- Euclidean distance NMI result: ***0.16870094590614582***
- L2 norm distance NMI result: ***0.16050194750614536***
- Euclidean distance NMI result(over 25 iterations): ***0.18742369451911783***

Since the value of NMI result using the Euclidean distance was low, my first thought was that the algorithm didn’t find the optimum. Then I run the algorithm with the L2 norm distance and the result wasn't that different and almost identical. I increased the number of iterations to 25 iterations and yet the algorithm didn't converge and the NMI results didn't increase that much hence I didn't reach a conclusion on whether the algorithm will ever converge, debugging and finding the issues was not feasible in my limited available time for this project.

Comparison: sklearn.cluster KMeans

- Runtime for the algorithm over 5 Iterations: ***597.1322*** seconds
- Achieved NMI over 5 iterations: ***0.8517726266820533***

Conclusion:
This implementation version of Kmeans algorithm is much faster than my implementation of Lloyd’s algorithm and much more accurate.

Task 2 and 3 were not implemented