1 Instructions

1. You can use any programming language of your choice with the suitable input format for each of the question. Each question should have a readme file and the detailed report.

2. A readme file should precisely tell how to compile and run your program. Give the exact commands with respect to the datasets provided.

3. The marks will be given on the basis of quality of code, use of innovative data structures, scalability, correctness, and completeness of the report.

4. You are supposed to submit the assignment on google classroom no later than **26th October 2020**. This is a strict deadline and any assignment submitted later will not be consider for evaluation unless you take prior permission (at least 4 days before the submission deadline).

Datasets to be used:

1. Iris Dataset (https://archive.ics.uci.edu/ml/datasets/iris)

2. Spiral Dataset (https://github.com/milaan9/Clustering-Datasets). Choose 2D spiral dataset of any format feasible to you.

Problem 1:

Implement the two clustering algorithms: [20 Marks]

1. K-means Algorithm

2. DB-Scan Algorithm

Compare the performance of each algorithm on the given datasets and prepare a detailed report containing the following:

- For both the datasets given, find the value of K that is best suited based on the sum of squared error metric.

- With the optimal value of K found in the previous part, compare the two algorithms in terms of sum of the squared error and time taken by both the algorithms. Which algorithm do you think is best suited for the given datasets.

.

2 Problem 2:

Transform the given datasets onto their first two principal components using principal component analysis (You can use sklearn.decomposition.PCA) and plot the entire data on 2-D plane. On this transformed dataset, implement the following two algorithms to compute the density estimates: [25 Marks]

- EM algorithm with the assumption of Gaussian Mixture model (Take the value of K same as the one obtained in the first part of question 1)
- Denclue

Plot the estimated density functions from your algorithm against the data points generated. Clearly explain Which algorithm do you thing represent your dataset best?