Tools and Techniques for Data Science
Assignment 4

Instructions:

- The aim of this assignment is to give you understanding of basic clustering and k-means.
- Submit a well documented and intuitive notebook
- Assignment-4 is due on 20-01-2022 (i.e. Thursday till 11:59:59pm)
- Discussion among students is allowed but the deliverable of the assignment should be individual.
- There will be no extension in time and no late submission will be accepted.
- There are no restrictions in the assignment. Do as you think is better but give valid reasoning.

Say you are given a data set where each observed example has a set of features, but has no labels. Labels are an essential ingredient to a supervised algorithm like Decision Tree or Random Forest, which learns a hypothesis function to predict labels given features. So we can't run supervised learning. What can we do?

One of the most straightforward tasks we can perform on a data set without labels is to find groups of data in our dataset which are similar to one another -- what we call clusters.

K-Means is one of the most popular "clustering" algorithms. K-means stores k centroids that it uses to define clusters. A point is considered to be in a particular cluster if it is closer to that cluster centroid than any other centroid.

K-Means finds the best centroids by alternating between (1) assigning data points to clusters based on the current centroids (2) choosing centroids (points which are the center of a cluster) based on the current assignment of data points to clusters.

Your task is to:

1. Generate synthetic data for k-means.
    a. Must have more than 10 dimensions
    b. Must be more than 5K rows
    c. A possible strategy could be to choose ‘k’ set of means and standard deviations and pass it to a random points generator. This would automatically create k

clusters corresponding to the means and standard deviations. Make sure that the means are not too far and not too close.

d. It’s better that you know the number of clusters initially for validation of results. However, it is not necessary to follow the above strategy. You could come up with a better solution.

2. Implement k-means algorithm on the dataset.
   a. You must code the algorithm yourself
   b. Use euclidean and manhattan distance as distance measure (both)
   c. You would have a viva or evaluation in which you would explain your algorithm, so make sure it is well commented and you understand it well.
3. Start with value of k=2 and increase the value much more than your original number of clusters
4. For each value of k, report SSE
5. Finally, draw a plot of SSE versus number of clusters (separately for euclidean and manhattan distance)

Try to interpret your graph and pay careful attention as to how the SSE changes before your original number of clusters and after it.

You can use numpy, pandas or basic libraries but do not use any external library for algorithms especially sklearn.