Kshitija Shelar | Text Analytics COMP4760 | Date: 15/11/20

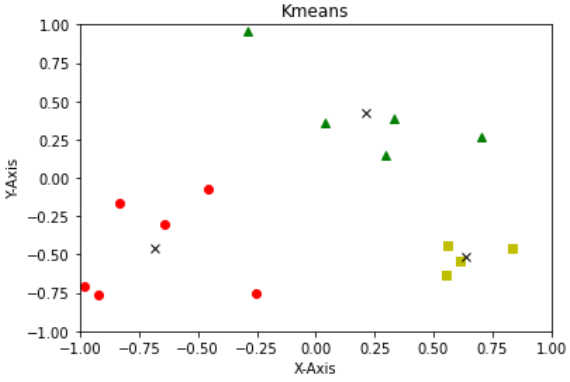
720200155 | Practical 7

1. K-means Clustering –

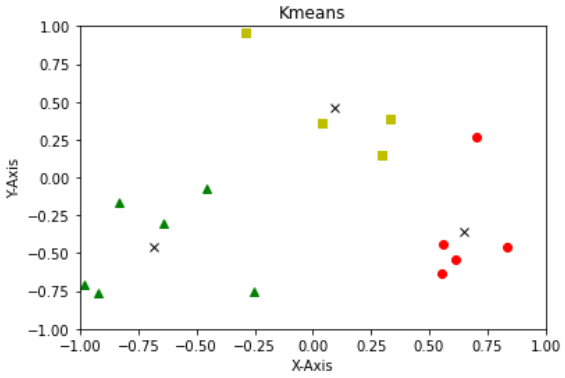
This is an unsupervised learning algorithm. Clustering is a process of grouping together points which occur in clusters. In k-means, the data points are classified under 'k' clusters. This usually works as choosing random points from the clusters and calculating distance of each point from these centre points. The points which are near are put in one cluster and the mean of the cluster is calculated. The distance of each point from this new mean is again calculated and this process repeats till desired result is obtained. For the task, following data points (15 random data points) are considered and plotted.

```
[[ 0.83577267 -0.46120404]
 [ 0.56189562 -0.43979864]
 [ 0.03744518  0.36043416]
 [-0.25348083 -0.74947791]
 [-0.92163114 -0.76121384]
 [-0.63833866 -0.30691994]
 [-0.83256332 -0.16902132]
 [ 0.33163975  0.39036818]
 [ 0.70014424  0.26487845]
 [ 0.61034854 -0.54116575]
 [ 0.29552672  0.14690514]
 [ 0.55131436 -0.63176047]
 [-0.28767824  0.95368876]
 [-0.98202436 -0.70664031]
 [-0.45661574 -0.06990991]]
```

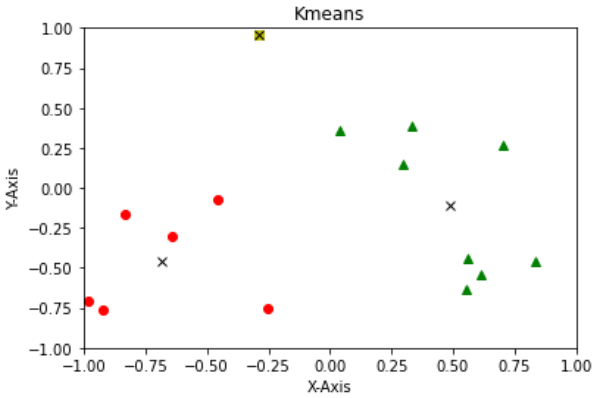
These data points are run 10 times and 4 observations are noted. The first cluster is formed 5/10 times, and the rest of the times other patterns are observed. Different patterns are observed because as explained above, every time a random centre point is selected and a new centroid is calculated. Because of this, as the centroid changes, the cluster changes.



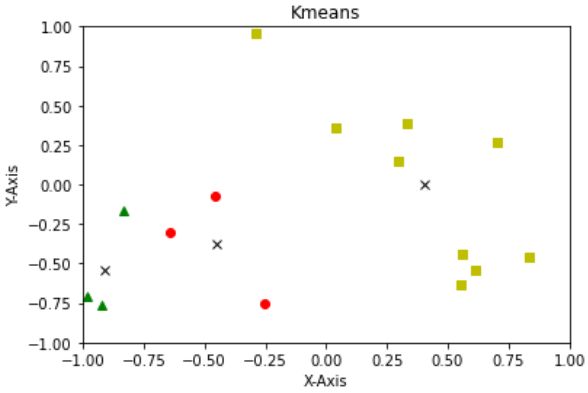
5/10 times



3/10 times



1/10 times

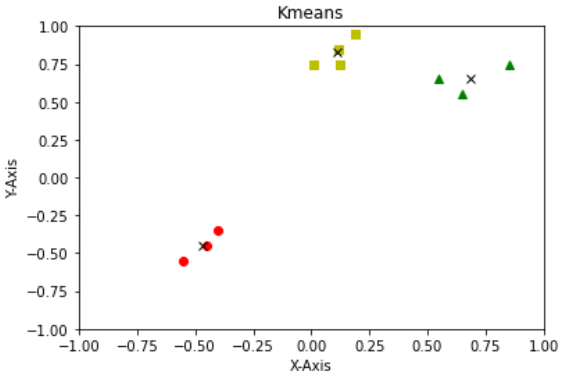


1/10 times

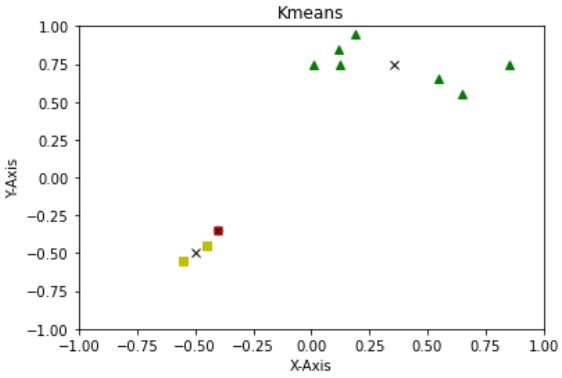
2. 10 data points are as follows:

```
[[ 0.55    0.65 ]
 [ 0.125   0.75 ]
 [ 0.65    0.55 ]
 [ 0.85    0.75 ]
 [-0.4    -0.35 ]
 [-0.45   -0.45 ]
 [-0.55   -0.55 ]
 [ 0.01    0.75 ]
 [ 0.12    0.85 ]
 [ 0.19    0.95 ]]
```

The clusters formed by these points are as follows:



8/10



2/10

It was observed that 1st image / cluster was formed 8 out of 10 times and the 2nd was formed 2 out of 10 times.

3 data points were modified such that they come closer to the centroid of their respective clusters. Points changed were

-0.4,-0.35 -> -0.4,-0.4

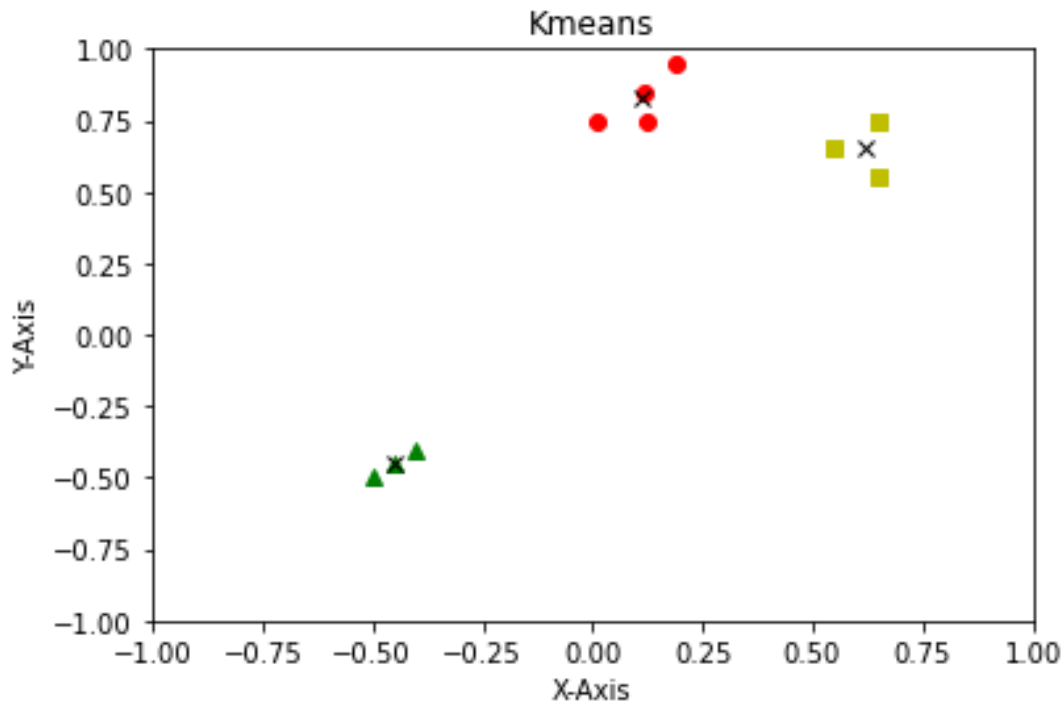
0.85,0.75 -> 0.65,0.75

-0.55,-0.55 -> -0.50,-0.50

```
[[ 0.55   0.65 ]
 [ 0.125  0.75 ]
 [ 0.65   0.55 ]
 [ 0.65   0.75 ]
 [-0.4   -0.4  ]
 [-0.45  -0.45 ]
 [-0.5   -0.5  ]
 [ 0.01   0.75 ]
 [ 0.12   0.85 ]
 [ 0.19   0.95 ]]
```

This change/modification of points had a huge impact. After this change, the clusters formed after 10 run times were the same. This is so because inter cluster distance is more and intra cluster distance is less. After modification, the points are all close to the centroid which is why 3 similar clusters are formed after every run.

Cluster after modification looks like follows:



3. We find different clusters in each run of kmeans algorithm because, random points are selected during initialization and the distance is calculated from these points to the rest of the points. Hence, the clusters formed would largely depend on these random points which are selected during initialization. The algorithm contains a few random points that uses same data points multiple time to find a global optimum solution. Because random points are selected, there is no consistency in the clusters formed. Kmeans algorithm is also sensitive to noisy data. [2]

   To improve the clusters, we can do better initialization and repeats of algorithm. [1][2]

   There is a method called simple furthest point heuristic which is used for initialization and this reduces clustering error by 15% - 6%. [2]

   It chooses the first centroid randomly and the next ones using a weighted probability pi = costi/sum(costi), where costi is the squared distance of the data point x_i to its nearest centroids.[2] It selects an arbitrary point as the first centroid and then adds new centroids one by one. At each step, the next centroid is the point that is furthest (max) from its nearest (min) existing centroid. This is also known as Maxmin[2].

   This technique helps to avoid worst case behaviour of the random centroids, especially when the cluster sizes have serious unbalance.[2]

References:

[1] https://www.researchgate.net/publication/1960044_Farthest-Point_Heuristic_based_Initialization_Methods_for_K-Modes_Clustering

[2] https://www.sciencedirect.com/science/article/pii/S0031320319301608