## *ASSIGNMENT NO.4*

**Title**: Write a program to do following:

We have given a collection of 8 points. P1=[0.1,0.6] P2=[0.15,0.71] P3=[0.08,0.9] P4=[0.16, 0.85] P5=[0.2,0.3] P6=[0.25,0.5] P7=[0.24,0.1] P8=[0.3,0.2]. Perform the k-mean clustering with initial centroids as m1=P1=Cluster#1=C1 and m2=P8=cluster#2=C2.

Answer the following:

a) Which cluster does P6 belong to?

b) What is the population of a cluster around m2?

c) What is the updated value of m1 and m2?

**Software/Libraries Used**:

- Python
- numpy
- matplotlib

**Theory/Methodology:**

K-Means clustering is an unsupervised machine learning algorithm used for partitioning data into k clusters. The algorithm works iteratively to assign each data point to the nearest centroid and then update the centroids based on the mean of the data points assigned to each cluster.

**Advantages:**

- Simple and easy to implement.
- Efficient for large datasets.
- Works well with data that is well-separated into clusters.

**Limitations/Examples:**

- Requires the number of clusters (k) to be specified in advance.

- Sensitive to the initial placement of centroids, which can affect the final clustering result.

- May not perform well with clusters of different sizes or densities.

**Working/Algorithm:**:

1. Initialize centroids (m1 and m2) based on given points P1 and P8.

2. Assign each point to the nearest centroid (cluster) based on Euclidean distance.

3. Calculate the mean of the points in each cluster to update the centroids.

4. Repeat steps 2 and 3 until convergence (centroids do not change significantly).

5. After convergence, determine:

   - Which cluster P6 belongs to.

   - Population of the cluster around m2.

   - Updated values of m1 and m2.

**Conclusion**:

K-Means clustering is a powerful algorithm for partitioning data into clusters based on similarity. In this practical, we implemented K-Means clustering on a collection of points and answered specific questions regarding cluster assignments, cluster populations, and centroid updates. This algorithm provides a straightforward approach to cluster analysis, but it's essential to consider its limitations and understand the impact of parameter choices on the clustering results.