St. Francis Institute of Technology, Mumbai-400 103

**Department Of Information Technology**

A.Y. 2024-2025

Class: TE-ITA/B, Semester: VI

Subject: **Business Intelligence Lab**

**Experiment – 6: To Implement K Means** **clustering algorithm using Java/Python**

1. **Aim: :** To Implement K Means clustering algorithm using Java/Python
2. **Objectives:** After study of this experiment, the students will be able to implement K Means Algorithm
3. **Outcomes:** After study of this experiment, the students will be able to

**CO 4:** Design andImplement various clustering data mining techniques such as Partitioning methods, Hierarchical Methods, Density - Based methods along with identification and analysisof outlier.

1. **Prerequisite:** Introduction to all the three clustering algorithms & Problem solving approach.
2. **Requirements:** Personal Computer, Windows XP operating system/Windows 7, Internet Connection, Microsoft Word, WEKA tool.
3. **Theory:**

**a. What is Clustering in Data Mining?**

Ans: Clustering in data mining is an unsupervised learning technique that groups similar data points into clusters based on their characteristics. It helps identify patterns and relationships in large datasets without predefined labels. Common types include K-Means (partitioning),Hierarchical, Density-Based (DBSCAN), and Model-Based (GMM) clustering. The K-Means algorithm assigns data points to the nearest centroid and iterates until convergence. Clustering is widely used in customer segmentation, image processing, anomaly detection, and document classification. It enhances data analysis by uncovering hidden structures, making it valuable in various industries.

**b. Explain K-means clustering algorithm?**

Ans: K-Means is a popular clustering algorithm that groups data into K clusters based on

similarity. It works by:

1. Selecting K initial centroids randomly.

2. Assigning each data point to the nearest centroid, forming clusters.

3. Updating centroids by computing the mean of assigned points.

4. Repeating steps 2 & 3 until centroids no longer change (convergence).

K-Means is efficient and widely used in customer segmentation, image processing, and anomaly

detection but struggles with non-spherical clusters and outliers.

* 1. **Laboratory Exercise:** Implementation of K means Classification Algorithm using Java/Python, Printout of implementation along with coding and snapshot

1. **Post-Experiments Exercise**
2. **Questions:**
   * In form of MCQ type test
   * K means solved numerical
3. **Conclusion:**
   * Summary of Experiment
   * Importance of Experiment
   * Application of Experiment
4. **Reference:** Data Mining: Concept & Techniques, 3rd Edition, Jiawei Han, Micheline  Kamber, Jian Pei, Elsevier.

**Laboratory Exercise:**

**Code:**

import random

def k\_means\_clustering():

data = list(map(int, input("Enter the dataset (comma-separated): ").split(',')))

k = int(input("Enter the number of clusters: "))

means = list(map(int, input(f"Enter {k} initial centroids (comma-separated): ").split(',')))

if len(means) != k:

print("Error: Number of initial centroids must match k.")

return

centroids = means

for iteration in range(1, 101):

clusters = [min(range(k), key=lambda i: abs(x - centroids[i])) for x in data]

new\_centroids = [round(sum(x for x, c in zip(data, clusters) if c == i) / clusters.count(i), 2) if clusters.count(i) > 0 else centroids[i] for i in range(k)]

for i in range(k):

assigned\_points = [data[j] for j in range(len(data)) if clusters[j] == i]

print(f"Iteration {iteration}: m{i+1}={new\_centroids[i]}")

print(f"k{i+1}={', '.join(map(str, assigned\_points))}")

if new\_centroids == centroids:

print("Convergence reached.")

break

centroids = new\_centroids

print(f"\nFinal Centroids: {new\_centroids}")

k\_means\_clustering()

**Output:**

