|  |  |
| --- | --- |
| Semester | T.E. Semester VI – Computer Engineering |
| Subject | Data Warehousing and Mining |
| Subject Professor In-charge | Prof. Kavita Shirsat |
| Assisting Teachers | Prof. Kavita Shirsat |
| Laboratory | Lab 312 A |

|  |  |  |
| --- | --- | --- |
| Student Name | Deep Salunkhe | |
| Roll Number | 21102A0014 | |
| Grade and Subject  Teacher’s Signature |  |  |

|  |  |  |
| --- | --- | --- |
| Experiment Number | 05 | |
| Experiment Title | K-Means Algorithm to find clusters | |
| Resources / Apparatus Required | Hardware:  Computer system | Software:  Python |
| Description | **Basic Idea:**   * The K-Means algorithm partitions a dataset into K distinct, non-overlapping clusters. * Each data point belongs to the cluster with the nearest mean (centroid).   **Steps of the K-Means Algorithm:**   1. **Initialization**: Choose K initial centroids. This can be done randomly or by selecting K data points from the dataset. 2. **Assignment**: Assign each data point to the nearest centroid. The assignment is typically based on a distance metric, such as Euclidean distance or Manhattan distance. 3. **Update Centroids**: Recalculate the centroids for each cluster as the mean (average) of all data points assigned to that cluster. 4. **Repeat**: Steps 2 and 3 are repeated until a stopping criterion is met. Common stopping criteria include a maximum number of iterations or when centroids no longer change significantly.   **Key Concepts:**   * **Centroid**: Each cluster is represented by its centroid, which is the mean of all data points in that cluster. * **Cluster Variance**: K-Means aims to minimize the within-cluster variance. It tries to ensure that data points within the same cluster are close to each other in terms of distance.   **Challenges and Considerations:**   * **Initial Centroid Selection**: The choice of initial centroids can affect the final clustering result. Different initializations may lead to different cluster assignments. * **Number of Clusters (K)**: You often need to specify the number of clusters (K) in advance. Selecting an inappropriate K value can result in suboptimal clustering. * **Convergence**: K-Means may converge to a local minimum, meaning it might not find the best clustering solution. Running the algorithm multiple times with different initializations can mitigate this issue. * **Scalability**: For large datasets, K-Means can be computationally expensive. There are variants like Mini-Batch K-Means for handling large datasets.   **Use Cases:**   * K-Means is commonly used for customer segmentation, image compression, anomaly detection, and recommendation systems.   **Advantages:**   * Simple and easy to implement. * Scales well to large datasets. * Often provides meaningful results.   **Disadvantages:**   * Requires specifying the number of clusters (K). * Sensitive to initial centroid selection. * May not perform well with non-spherical or unevenly sized clusters. | |
| Program |  | |
| Output |  | |
| Conclusion: | K-Means is a versatile and widely used clustering algorithm that finds natural groupings in data by iteratively optimizing cluster centroids. While it has its limitations, it remains a valuable tool in data analysis and machine learning. | |