MACHINE LEARNING ASSIGNMENT
PROGRAM ELECTIVE IV

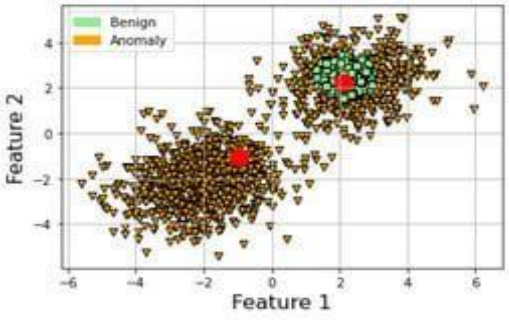
NAME: Anshu Kumar Bishwas
REGISTRATION NUMBER: 202000284

Objective: Prepare PPT/notes on K-Means Clustering and Agglomerative hierarchical Clustering Algorithm. Show examples/working for each algorithm.

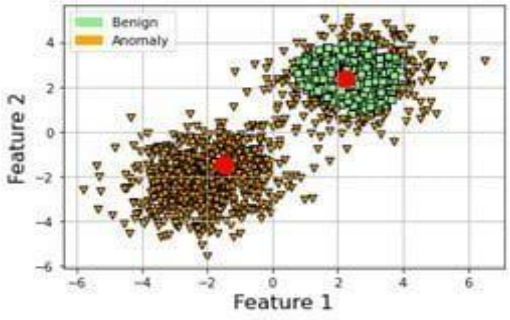
K-Means Clustering:

1. K-Means is an unsupervised clustering algorithm that aims to partition data into K clusters, where each data point belongs to the cluster with the nearest mean.
2. **Algorithm**:
    - Initialize K cluster centroids randomly.
    - Assign each data point to the nearest centroid.
    - Recalculate the centroids as the mean of points in each cluster.
    - Repeat the assignment and centroid update steps until convergence (when centroids no longer change significantly).

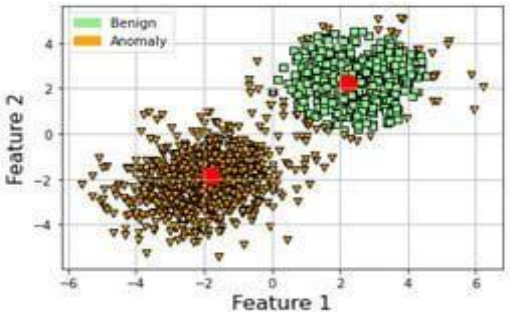
3. **Example:**



BaselineKmeans with percentile value of 90%



BaselineKmeans with percentile value of 95%



BaselineKmeans with percentile value of 98%

Step 1: Initialize Centroids

Step 2: Assign Points to Nearest Centroids

Point A is closer to Centroid 1, so it is assigned to Cluster 1.

Point B is closer to Centroid 1, so it is assigned to Cluster 1.

Point C is closer to Centroid 2, so it is assigned to Cluster 2.

Step 3: Recalculate Centroids

New Centroid 1 is the average of points in Cluster 1: [(1, 2) + (4, 5)] / 2 = (2.5, 3.5) New Centroid 2 is the average of points in Cluster 2: (8, 8)

Step 4: Repeat until Convergence

Repeat steps 2 and 3 until the centroids no longer change significantly. In this case, the centroids have converged, and the final clusters are:

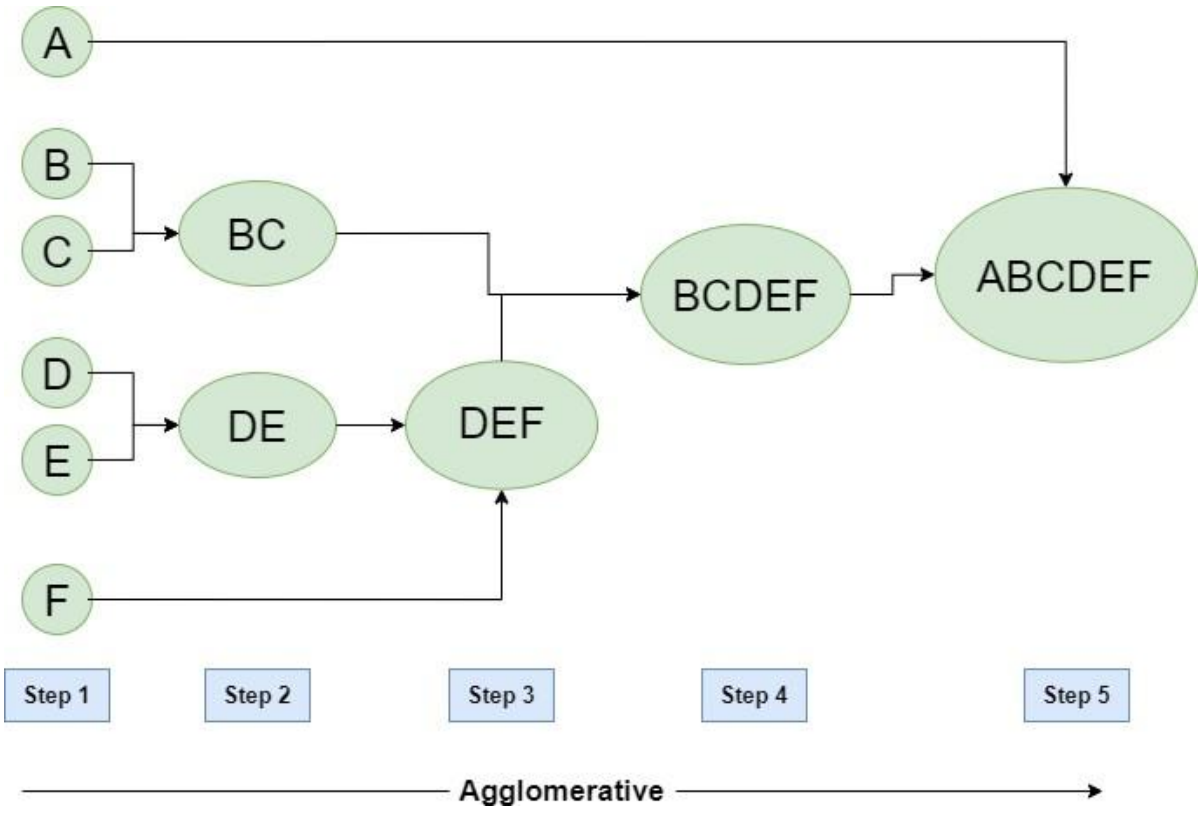
Cluster 1: Points A and B

Cluster 2: Point C

4. **Applications**: Image compression, customer segmentation, anomaly detection.

Agglomerative Hierarchical Clustering:

1. Agglomerative hierarchical clustering is a bottom-up approach to clustering. It starts with individual data points as separate clusters and merges them until all points belong to a single cluster or a predefined number of clusters is reached.
2. **Algorithm**:
    - Start with each data point as a separate cluster.
    - Merge the two closest clusters into a new cluster.
    - Repeat the merging process until the desired number of clusters is achieved or all points belong to a single cluster.
3. **Example:**



Steps:

- Consider each alphabet as a single cluster and calculate the distance of one cluster from all the other clusters.
- In the second step, comparable clusters are merged together to form a single cluster. Let's say cluster (B) and cluster (C) are very similar to each other therefore we merge them in the

second step similarly to cluster (D) and (E) and at last, we get the clusters [(A), (BC), (DE), (F)]

- We recalculate the proximity according to the algorithm and merge the two nearest clusters([(DE), (F)]) together to form new clusters as [(A), (BC), (DEF)]
- Repeating the same process; the clusters DEF and BC are comparable and merged together to form a new cluster. We're now left with clusters [(A), (BCDEF)].
- At last, the two remaining clusters are merged together to form a single cluster [(ABCDEF)].

4. **Applications**: Phylogenetic, taxonomy, image segmentation.

Both K-Means and Agglomerative Hierarchical Clustering have their strengths and weaknesses. K-Means is a partitional method that requires specifying the number of clusters in advance, while Agglomerative Hierarchical Clustering builds a tree-like structure that doesn't require specifying the number of clusters beforehand. The choice of algorithm depends on the nature of your data and the specific goals of your analysis.