| Experiment No.8 |
| --- |
| Clustering algorithm in Big data |
| Date of Performance: |
| Date of Submission: |

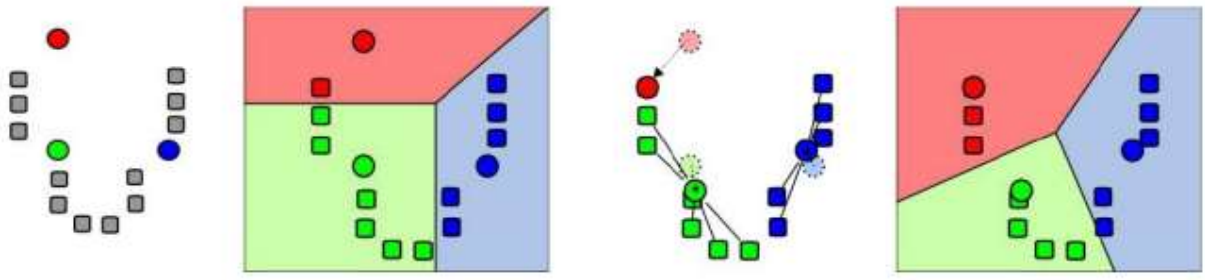
EXPERIMENT NO. 08

Aim: To study clustering algorithm in Big data

Theory:
What is Clustering?
• Generally, a group of abstract objects into classes of similar objects is made.
• We treat a cluster of data objects as one group.
• While doing cluster analysis, we first partition the set of data into groups. That based on data similarity and then assign the labels to the groups.
• The main advantage of over-classification is that it is adaptable to changes. And helps single out useful features that distinguish different groups.
• Data Clustering analysis is used in many applications. Such as market research, pattern recognition, data analysis, and image processing.
• Data Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.

Clustering is a Machine Learning technique that involves the grouping of data points. Given a set of data points, we can use a clustering algorithm to classify each data point into a specific group. In theory, data points that are in the same group should have similar properties and/or features, while data points in different groups should have highly dissimilar properties and/or features. Clustering is a method of unsupervised learning and is a common technique for statistical data analysis used in many fields.

There are many different clustering models:
• Connectivity models based on connectivity distance.
• Centroid models based on central individuals and distance.
• Density models based on connected and dense regions in space.
• Graph-based models based on cliques and their relaxations.
The simplest clustering algorithm is k-means, which is a centroid-based model. Shown in the images below is a demonstration of the algorithm



We start out with k initial "means" (in this case, k = 3), which are randomly generated within the data domain (shown in color). k clusters are then created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means. The centroid of each of the k clusters becomes the new mean. Steps 2 and 3 are repeated until convergence has been reached.

K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells.

Given a set of observations (x1, x2, …, xn), where each observation is a d-dimensional real vector, k-means clustering aims to partition the n observations into k groups G = {G1, G2, …, Gk} so as to minimize the within-cluster sum of squares (WCSS) defined as follows −

argmin∑i=1k∑x∈Si‖x−μi‖2argmin∑i=1k∑x∈Si‖x−μi‖2

The later formula shows the objective function that is minimized in order to find the optimal prototypes in k-means clustering. The intuition of the formula is that we would like to find groups that are different with each other and each member of each group should be similar with the other members of each cluster.

The following example demonstrates how to run the k-means clustering algorithm in R.

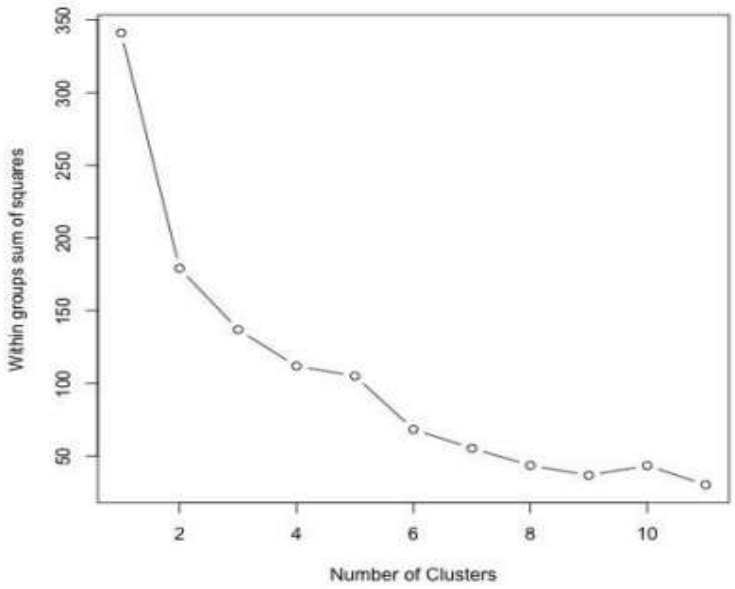
```
library(ggplot2)
# Prepare Data
data = mtcars

# We need to scale the data to have zero mean and unit variance
data <- scale(data)

# Determine number of clusters
wss <- (nrow(data)-1)*sum(apply(data,2,var))
for (i in 2:dim(data)[2]) {
 wss[i] <- sum(kmeans(data, centers = i)$withinss)
}


# Plot the clusters
plot(1:dim(data)[2], wss, type = "b", xlab = "Number of Clusters",
 ylab = "Within groups sum of squares")
```

In order to find a good value for K, we can plot the within groups sum of squares for different values of K. This metric normally decreases as more groups are added, we would like to find a point where the decrease in the within groups sum of squares starts decreasing slowly. In the plot, this value is best represented by K = 6.




Now that the value of K has been defined, it is needed to run the algorithm with that value.

```
# K-Means Cluster Analysis
fit <- kmeans(data, 5) # 5 cluster solution

# get cluster means
aggregate(data,by = list(fit$cluster),FUN = mean)

# append cluster assignment
data <- data.frame(data, fit$cluster)
```

**Conclusion :**

Clustering is a Machine Learning technique that involves the grouping of data points. Given a set of data points, we can use a clustering algorithm to classify each data point into a specific group. In theory, data points that are in the same group should have similar properties and/or features, while data points in different groups should have highly dissimilar properties and/or features. Clustering is a method of unsupervised learning and is a common technique for statistical data analysis used in many fields.