**Project Based Learning Report**

on

**K-means Clustering**

Submitted in the partial fulfillment of the requirements

For the Project based learning in (**Fuzzy logic, Neural Network and Genetic Algorithm** )

in

Electronics & Communication Engineeringb

By

**2014111114 VAISHNAVI SINGH**

**2014111118 SAHIL SINHA**

**2014111119 SHASHANK KUMAR SRIVASTAV**

Under the guidance of Course In-charge

**Prof. – V.P. KADUSKAR**

Department of Electronics & Communication Engineering

Bharati Vidyapeeth

(Deemed to be University)

College of Engineering,

Pune – 411043

**Academic Year: 2022-23**

**Bharati Vidyapeeth**

**(Deemed to be University)**

**College of Engineering,**

**Pune – 411043**

**DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING**

**CERTIFICATE**

Certified that the Project Based Learning report entitled, “**BASIC SWITCHING THEORY”** is work done by

**2014111114 VAISHNAVI SINGH**

**2014111118 SAHIL SINHA**

**2014111119 SHASHANK KUMAR SRIVASTAV**

in partial fulfillment of the requirements for the award of credits for Project Based Learning (PBL) in **Fuzzy Logic, Neural Network and Genetic Algorithm** of Bachelor of Technology Semester V, in ELECTRONICS AND COMMUNICATION ENGINEERING

**Date:**

**Prof. – V.P. KADUSKAR Dr. Arundhati A. Shinde**

**Course In-charge Professor & Head**

INDEX

|  |  |  |
| --- | --- | --- |
| **SR.NO** | **TITTLE** | **PAGE NO.** |
| **1** | **Problem Statement of the project Based Learning:** | 4-4 |
| **2** | **Theory** | 5-8 |
| **3** | **Software** | 9 |
| **4** | **Simulation and Outcome** | 10 |
| **5** | **Project outcome and conclusion** | 11-12 |

**Q) Problem Statement-**

**What is K means clustering and how can you implement it using MATLAB?**

**Solution-**

K-Means Clustering Algorithm

K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or data science. In this topic, we will learn what is K-means clustering algorithm, how the algorithm works, along with the Python implementation of k-means clustering.

What is K-Means Algorithm?

K-Means Clustering is an [Unsupervised Learning algorithm](https://www.javatpoint.com/unsupervised-machine-learning), which groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on.

It allows us to cluster the data into different groups and a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training.

It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.

The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.

The k-means [clustering](https://www.javatpoint.com/clustering-in-machine-learning) algorithm mainly performs two tasks:

* Determines the best value for K center points or centroids by an iterative process.
* Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.

Hence each cluster has datapoints with some commonalities, and it is away from other clusters.

The below diagram explains the working of the K-means Clustering Algorithm:



How does the K-Means Algorithm Work?

The working of the K-Means algorithm is explained in the below steps:

**Step-1:** Select the number K to decide the number of clusters.

**Step-2:** Select random K points or centroids. (It can be other from the input dataset).

**Step-3:** Assign each data point to their closest centroid, which will form the predefined K clusters.

**Step-4:** Calculate the variance and place a new centroid of each cluster.

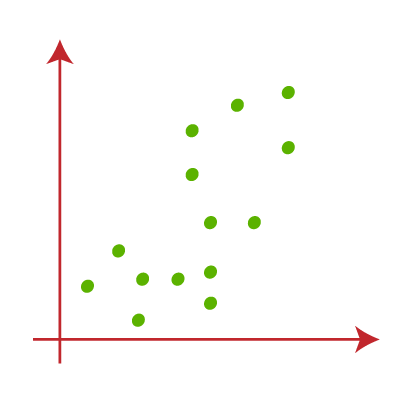
**Step-5:** Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

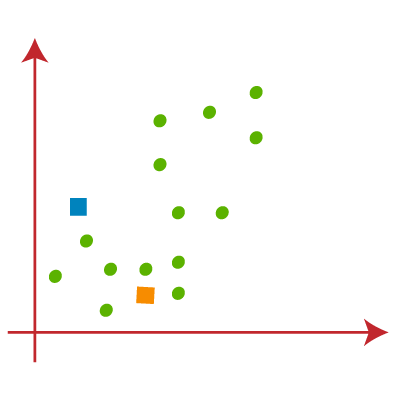
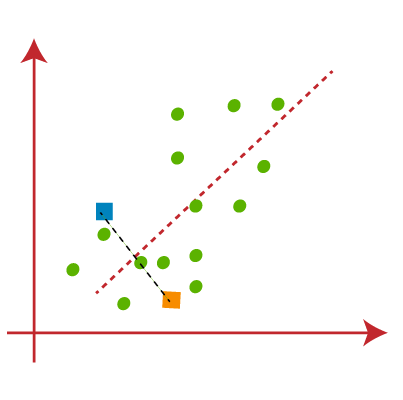
**Step-6:** If any reassignment occurs, then go to step-4 else go to FINISH.

**Step-7**: The model is ready.

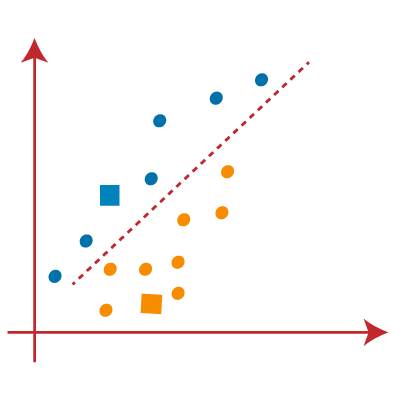
Let's understand the above steps by considering the visual plots:

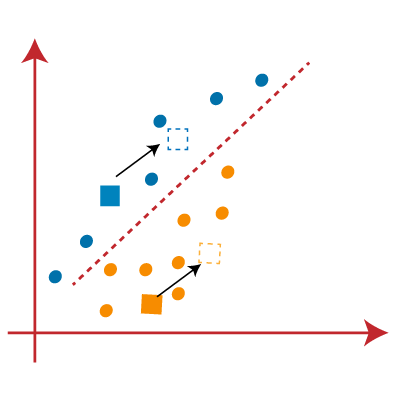
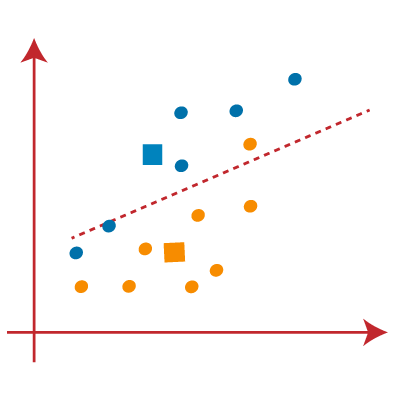
Suppose we have two variables M1 and M2. The x-y axis scatter plot of these two variables is given below:



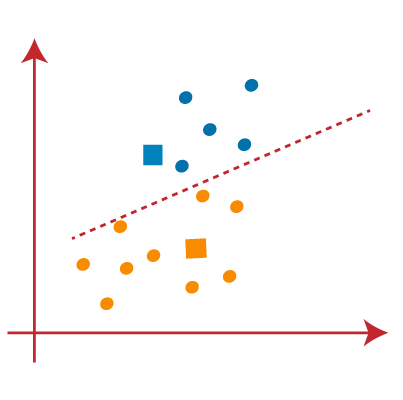
* Let's take number k of clusters, i.e., K=2, to identify the dataset and to put them into different clusters. It means here we will try to group these datasets into two different clusters.
* We need to choose some random k points or centroid to form the cluster. These points can be either the points from the dataset or any other point. So, here we are selecting the below two points as k points, which are not the part of our dataset. Consider the below image:  
  
* Now we will assign each data point of the scatter plot to its closest K-point or centroid. We will compute it by applying some mathematics that we have studied to calculate the distance between two points. So, we will draw a median between both the centroids. Consider the below image:  
  

From the above image, it is clear that points left side of the line is near to the K1 or blue centroid, and points to the right of the line are close to the yellow centroid. Let's color them as blue and yellow for clear visualization.

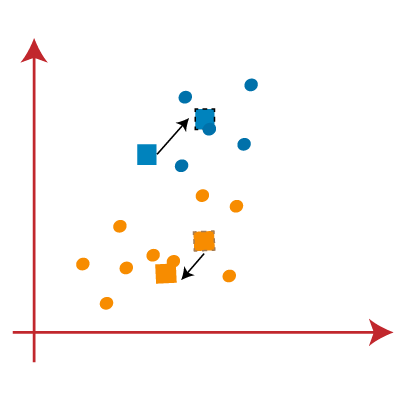
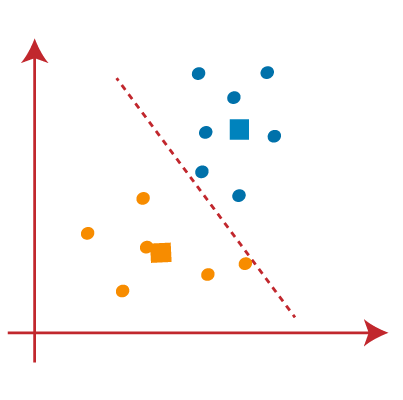
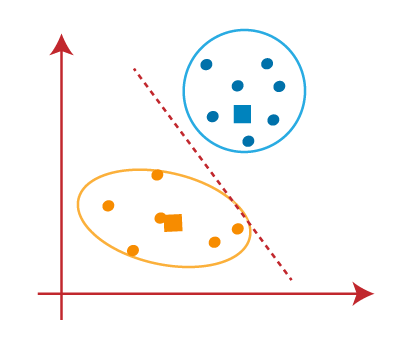


* As we need to find the closest cluster, so we will repeat the process by choosing **a new centroid**. To choose the new centroids, we will compute the center of gravity of these centroids, and will find new centroids as below:  
  
* Next, we will reassign each datapoint to the new centroid. For this, we will repeat the same process of finding a median line. The median will be like below image:  
  

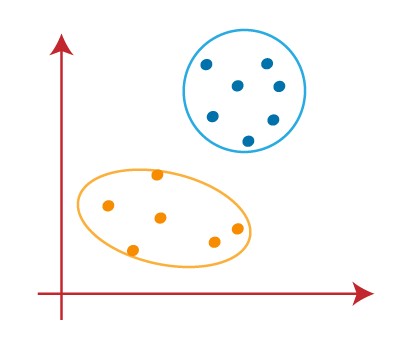
From the above image, we can see, one yellow point is on the left side of the line, and two blue points are right to the line. So, these three points will be assigned to new centroids.



As reassignment has taken place, so we will again go to the step-4, which is finding new centroids or K-points.

* We will repeat the process by finding the center of gravity of centroids, so the new centroids will be as shown in the below image:  
   
* As we got the new centroids so again will draw the median line and reassign the data points. So, the image will be:  
   
* We can see in the above image; there are no dissimilar data points on either side of the line, which means our model is formed. Consider the below image:  
   

As our model is ready, so we can now remove the assumed centroids, and the two final clusters will be as shown in the below image:



## How to choose the value of "K number of clusters" in K-means Clustering?

The performance of the K-means clustering algorithm depends upon highly efficient clusters that it forms. But choosing the optimal number of clusters is a big task. There are some different ways to find the optimal number of clusters, but here we are discussing the most appropriate method to find the number of clusters or value of K. The method is given below:

### Elbow Method

The Elbow method is one of the most popular ways to find the optimal number of clusters. This method uses the concept of WCSS value. **WCSS** stands for **Within Cluster Sum of Squares**, which defines the total variations within a cluster. The formula to calculate the value of WCSS (for 3 clusters) is given below:

WCSS= ∑Pi in Cluster1 distance(Pi C1)2 +∑Pi in Cluster2distance(Pi C2)2+∑Pi in CLuster3 distance(Pi C3)2

In the above formula of WCSS,

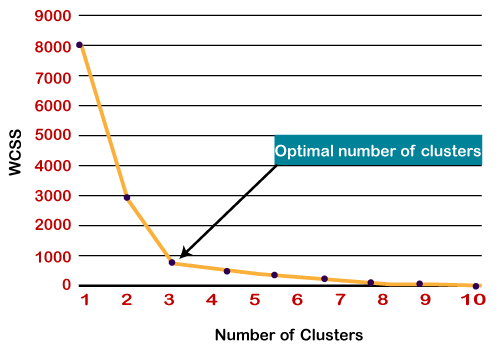
∑Pi in Cluster1 distance(Pi C1)2: It is the sum of the square of the distances between each data point and its centroid within a cluster1 and the same for the other two terms.

To measure the distance between data points and centroid, we can use any method such as Euclidean distance or Manhattan distance.

To find the optimal value of clusters, the elbow method follows the below steps:

* It executes the K-means clustering on a given dataset for different K values (ranges from 1-10).
* For each value of K, calculates the WCSS value.
* Plots a curve between calculated WCSS values and the number of clusters K.
* The sharp point of bend or a point of the plot looks like an arm, then that point is considered as the best value of K.

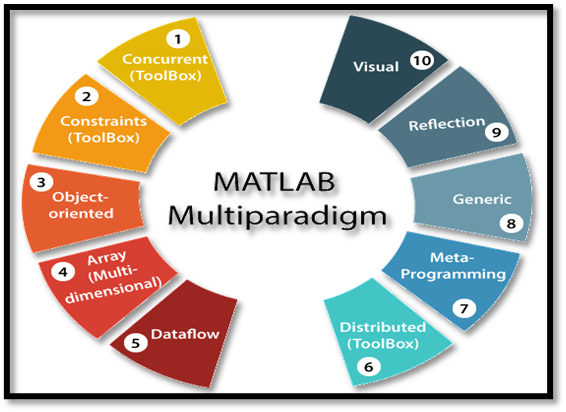
Since the graph shows the sharp bend, which looks like an elbow, hence it is known as the elbow method. The graph for the elbow method looks like the below image:



**SOFTWARE**

**MATLAB INTRODUCTION**

* MATLAB was invented by mathematician and computer programmer [Cleve](https://en.wikipedia.org/wiki/Cleve_Moler) Moler.The idea for MATLAB was based on his 1960s PhD thesis.Moler became a math professor at the University of New Mexico and started developing MATLAB for his students as a hobby. He developed MATLAB's initial linear algebra programming in 1967 with his one-time thesis advisor, [George](https://en.wikipedia.org/wiki/George_Forsythe) Forsythe. This was followed by [Fortran](https://en.wikipedia.org/wiki/Fortran) code for linear equations in 1971.
* In the beginning (before version 1.0) MATLAB "was not a programming language; it was a simple interactive matrix calculator. There were no programs, no toolboxes, no graphics. And no ODEs or [FFTs](https://en.wikipedia.org/wiki/FFT)."
* The first early version of MATLAB was completed in the late 1970s. The software was disclosed to the public for the first time in February 1979 at the [Naval Postgraduate School](https://en.wikipedia.org/wiki/Naval_Postgraduate_School) in California.Early versions of MATLAB were simple [matrix calculators](https://en.wikipedia.org/wiki/Matrix_(mathematics)) with 71 pre-built functions. At the time, MATLAB was distributed for free to universities. Moler would leave copies at universities he visited, and the software developed a strong following in the math departments of university campuses.



**Development Environment:-** This is the set of tools and facilities that help you use MATLAB operations and files. Many of these tools are the graphical user interface. It involves the MATLAB desktop and command window, a command history, an editor and debugger, and browsers for considering help, the workspace, reports, and the search path. **MATLAB Mathematical Function Library.** This is a vast compilation of computing design ranging from basic functions, like sum, sine, cosine, and complex mathematic, to more sophisticated features like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

**MATLAB Language:-** This is a high level matrix/array language with control flow statement, function, data structure, input/output, and object-oriented programming characteristics. It allows both "programming in the small" to create quick and dirty throw-away programs rapidly and "programming in the large" to create large and complex application functions.

**MATLAB External Interfaces/API:-** This is a library that allows us to write C and FORTRAN programs that interact with MATLAB. It contains facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files

**Simulation:-**

This is the code for the simulation which is written in MATLAB

%%

load fisheriris

X = meas(:,3:4);

figure;

plot(X(:,1),X(:,2),'k\*','MarkerSize',5);

title 'Fisher''s Iris Data';

xlabel 'Petal Lengths (cm)';

ylabel 'Petal Widths (cm)';

%%

rng(1); % For reproducibility

[idx,C] = kmeans(X,3);

%%

x1 = min(X(:,1)):0.01:max(X(:,1));

x2 = min(X(:,2)):0.01:max(X(:,2));

[x1G,x2G] = meshgrid(x1,x2);

XGrid = [x1G(:),x2G(:)]; % Defines a fine grid on the plot

idx2Region = kmeans(XGrid,3,'MaxIter',1,'Start',C);

%%

figure;

gscatter(XGrid(:,1),XGrid(:,2),idx2Region,...

[0,0.75,0.75;0.75,0,0.75;0.75,0.75,0],'..');

hold on;

plot(X(:,1),X(:,2),'k\*','MarkerSize',5);

title 'Fisher''s Iris Data';

xlabel 'Petal Lengths (cm)';

ylabel 'Petal Widths (cm)';

legend('Region 1','Region 2','Region 3','Data','Location','SouthEast');

hold off;

**OUTCOME: -**

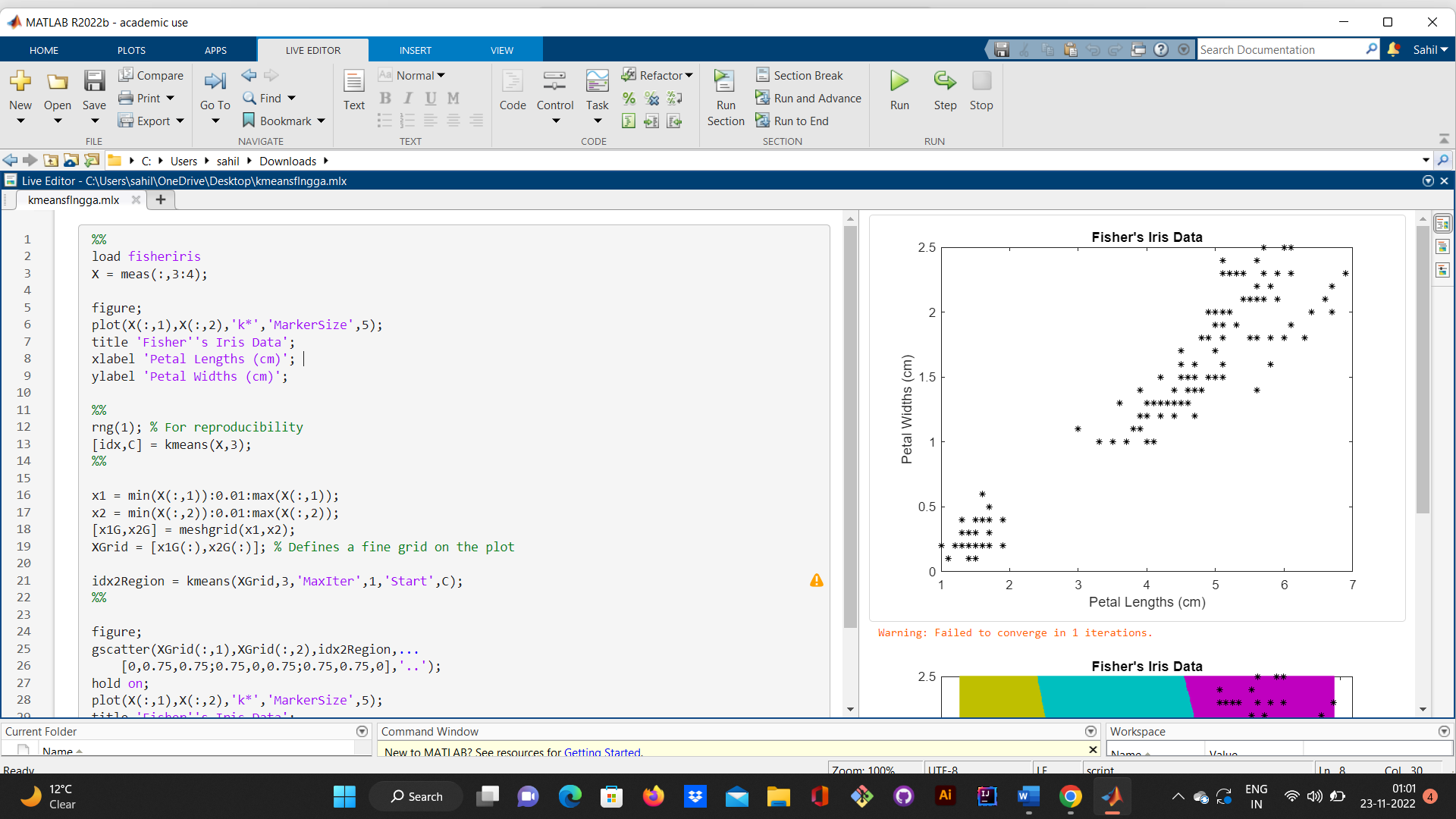


FIG-1(OUTPUT SCREENSHOT-1)

Graphical user interface, application

Description automatically generated

FIG-2(OUTPUT SCREENSHOT-2)

**Project Outcome and Conclusion**

**Project Outcome:** Through this following project we conclude the outcome that K means clustering is one of the most important methods of Unsupervised learning which is very helpful in Machine Learning and various different applications

**Conclusion**: So we can conclude that K means clustering is an important and very useful process of Unsupervised learning which can be very useful in day to day life as well as in various fields such as Machine Learning, Artificial Intelligence, Technology, Medical, etc. and many such various fields