KMEANS CLUSTERING

# Heading: To Find Out the clusters by using K MEANS CLUSTERING

## Datatypes: Here the data is continuous

### Syntax:

**Install.packages(“plyr) #Installing the packages from graphical representation**

**Library(plyr) #Loading the functions in library lattice**

**?plyr # To understand what all functions are included in plyr**

**View(input) # After importing the data to check all the columns**

**mydata<-input[1:25,c(2,3:8)] # Removing the unwanted columns**

**view(mydta) # After removing to check all the columns**

**normalizeddata<-scale(mydata[2:7]) # Normalizing the data by using scaling**

**view(normalizied data) #After normalizing to check the columns**

**km<-kmeans(normalizeddata,4) # By using the KMEANS Algorighm we have to normalize it**

**view(km) # After clustering we have to check the columns**

**str(km) # For calculating the output**

List of 9

$ cluster : int [1:25] 3 3 4 3 4 3 3 4 3 3 ...

$ centers : num [1:4, 1:6] -1.933 -1.767 0.8 -0.124 -2.117 ...

..- attr(\*, "dimnames")=List of 2

.. ..$ : chr [1:4] "1" "2" "3" "4"

.. ..$ : chr [1:6] "SAT" "Top10" "Accept" "SFRatio" ...

$ totss : num 144

$ withinss : num [1:4] 3.63 0 18.01 17.78

$ tot.withinss: num 39.4

$ betweenss : num 105

$ size : int [1:4] 3 1 11 10

$ iter : int 2

$ ifault : int 0

- attr(\*, "class")= chr "kmeans"

**Install.packages(“animation”) #**