Assignment no A6
Roll no 4202

1 Title

K-Means Clustering

2 Problem Definition

Implement a simple approach for k-means/k-medoids clustering using C++

3 Objective

- To understand various techniques of clustering.
- To implement k-means clustering using C++.

4 Pre-requisite

- Basic knowledge of object oriented programming
- Knowledge of basic clustering approach

5 Software and Hardware Requirements

1. 64 bit Machine i3/i5/i7
2. 64-bit open source Linux OS Fedora 20
3. g++ library
4. Eclipse

Theory

Clustering :

Clustering is the classification of objects into different groups, or more precisely, the partitioning of a data set into subsets (clusters), so that the data in each subset (ideally) share some common trait - often according to some defined distance measure.

Types of Clustering :

- Hierarchical algorithms: these find successive clusters using previously established clusters.
    1. Agglomerative ("bottom-up"): Agglomerative algorithms begin with each element as a separate cluster and merge them into successively larger clusters.
    2. Divisive ("top-down"): Divisive algorithms begin with the whole set and proceed to divide it into successively smaller clusters.
- Partitional clustering: Partitional algorithms determine all clusters at once.
  They include:
    1. K-Means
    2. K-Medoids

K-Means Method :

- K-Means is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem.
- The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori.
- The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other.
- The next step is to take each point belonging to a given data set and associate it to the nearest center.
- When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycenter of the clusters resulting from the previous step.
- After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center.
- A loop has been generated. As a result of this loop we may notice that the k centers change their location step by step until no more changes are done or in other words centers do not move any more.

Finally, this algorithm aims at minimizing an objective function know as squared error function given by:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|^2,$$

The k-means procedure is summarized in following figure. :

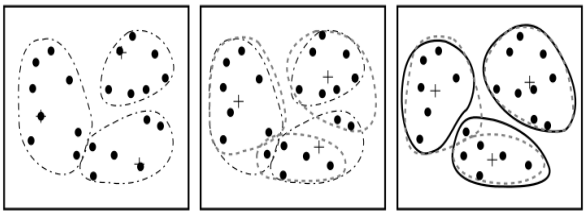


Figure Clustering of a set of objects based on the *k*-means method. (The mean of each cluster is marked by a "+".)

Algorithm:

Input :
k : the number of the clusters
D : a data set containing n objects.

Output : A set of k clusters.

Method :

1. arbitrarily choose k - objects from D as the initial cluster centers
2. repeat
3. (re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the clusters
4. update the cluster means, i.e.,calculate the mean value of the objects for each cluster ;
5. until no change.

6 Mathematical Model :

Let S be the solution perspective of the system such that,
S ={S_t,E_t,I,O,DD,NDD,F_{me},Sc,Fc}
where,

S_t = start state .

I represents set of input
I = {x,y}
where,
x represents the value of X-axis
y represents the value of Y-axis
(x,y) ∈ I+

O represents set of output
O = {c[]}
Where, c[] = array of nodes having type of class

F_{me}=set of functions.
F_{me}={f1,f2,f3,f4}
where,
f1= f1 represents the function to read the input.
f2= f2 represents the function to create a new cluster.
f3= f3 represents the function to calculate new mean values.
f4 = f4 to prints the cluster classes

E_t=end state
=display clusters.

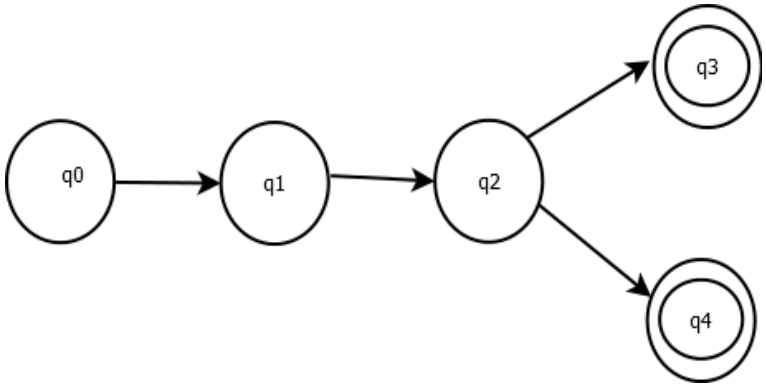
DD (Deterministic Data) = x and y coordinates are given by the user

NDD (Non Deterministic Data) = the cluster creation

Sc = data is classified according to knn algorithm.

Fc = data isnt classified, program in infinite loop

7 State Diagram :



q0 → start state. Accept the x and y coordinates

q1 → create clusters

q2 → to find the new mean value

q3 → to display if the cluster is created.

q4 → to display if the cluster is not created.

8 Conclusion

We have thus understood the concept of clustering and successfully implemented the k-mean algorithm.