Lines = n of lines of the dataset

k = n of clusters

Centroids = select k random points;

Convergence = some high value;

WHILE (Convergence > *B*)

Iterate over lines

Iterate over the number of cluster

Calculate the distance between the point and the centroid

Store in a list distance

Append distance to distances

##Create a placeholder in the dataset for the cluster number, in the last position

append (lines[i]) = -1

Iterate over distances

## Select the larger value in the distance[i]

Create a copy of the line distance Temp[] = distance[i]

sort(temp[]) ##biggest value is the last

Attribution = -1

Iterate over distance

If temp[0] == distance[j]

Attribution = j

Else attribution = attribution

Attribute the line to a cluster to dataset[i][columns] = Attribution

##Update the centroids (average of all points in the cluster)

Count[] = k positions at 0

Iterate over lines

Iterate over k

if (lines[columns] == k)

Iterate over every position in line[]

Sum[k][j] = sum1[j] + line[i][j] ;

Count[k]++;

##Calculate average distance

Iterate over sum

Iterate over positions in sum

Average[k][j] = sum[k][j] / Count[k]