Semester I Project

Project title

Research Project

Algorithm to find initial points for k means clustering method

Developed By:-

Disha Shah – 15

Umang Shah – 16

Poras Vyas - 18

M.Sc. (Artificial Intelligence and machine learning)

Presented to:-

Department of computer science,

Rollwala computer centre,

Gujarat University

Acknowledgement

We would like to thank our project Guide and Head of the Department of Computer Science, Gujarat University, Dr. Savita Gandhi and Professor Trushali Jambudi from Ahmedabad university for giving us the opportunity of working on this project and sharing this algorithm with us and helping us in this research by giving her algorithm and guiding us whenever we were stuck while working on the algorithm.

This project would not be a success without guidance of Dr. Savita Gandhi and Professor Trushali Jambudi, so we are thankful to them for helping and guiding us for this project and helping us to learn new things.

Thank You All

Disha Simit Shah

Umang Amit Shah

Poras Achal Vyas

Table of Contents

PROJECT PROFILE

Project Title:

Research project – clustering algorithm for finding initial centroids for k-means algorithm

Project team

Roll number	**Members' name**
15	Disha Simit Shah
16	Umang Amit Shah
18	Poras Achal Vyas

Project definition:

This project aims to find an algorithm for initializing the Initial Cluster centers and automatically determining the number of clusters to be formed in the original K-means algorithm.

Language used :	R language
Project guide :	Dr. Savita Gandhi
External Guide :	Prof. Trushali Jambudi (Ahmedabad university)
Submitted to :	Department of computer science ,Rollwala computer center

Objective

Our objective to apply this algorithm was to find initial centres for k-means algorithm, as every time we calculate k-means we need to give total number of cluster we want from data, result generate from k-means algorithm was random as centres where elected on the basis of mean of data every time clusters formed were different in rare cases they used to be similar.

So by proposing this algorithm, main idea was to overcome this problem and give good centres so our final outcome improves and we get good clusters based on the density of data centres are selected

Introduction:

The algorithm being explored is about reducing convergence time and increasing the quality of clustering by K-means clustering algorithm.

Clustering means grouping of a particular set of objects based on their characteristics, aggregating them according to their similarities. Clusters are formed such that data in one cluster is similar to each other but totally different from other data clusters.

It is used to find similarities in data. So it can help to categorize or classify data on basis of various measures.

K-means is one of the simplest unsupervised learning clustering algorithms that solve the well known clustering problem.

The main idea is to define 'k' no. of centres, one for each cluster.

Some problems with K-means are:

Need to specify k, the number of clusters, in advance

Performance is dependent on selection of initial centroids

Here you will be explained an algorithm which is supposed to help the k- means algorithm in finding the clusters faster and more acccurately, by providing the number of clusters to be formed and initial points to start k- means.

We plan on doing this with the use of positional measures of central tendency, quartiles.

Algorithm

1. Compute the Five number summary on the data.
2. Ignore the min and max value and select the rest of the 3 values as initial 3 centres i.e.
    a. the 25th percentile or lower quartile or first quartile
    b. the 50th percentile or median (middle value, or second quartile)
    c. the 75th percentile or upper quartile or third quartile
3. Determine the radius of the centres and form the bin around this radius.
4. Determine the density of each bin formed.
5. For the bins with density > minD,
    a. Add the centres to the list of centres
    b. Remove the data points falling in the bins' radius from the original data set.
6. Repeat steps 1 to 5 until centres are added to the centre list.
7. Check the distance between these centres if dist< lower 90% replace such centres by their mean as the centre value.

Forming the bins:

1. Computing Bin Density A bins' Density is the number of points in its radius
2. Determining minD As per han and kamber initial k can be: N/2 in expectation that each cluster has> V2N data points For unknown k, from above, each bin should have N data points
3. Determining Distance Radius
    a. For each bin we have the min, max and median/ centre point
    b. Compute the distance of each point in the bin with the min, max and centre value of that bin. Will work also in case of Method I if we set bin range such that eg Bin 1- Min, 10% as centre, 25% as max and so on for each in.
    c. Assign the data point to the radius of the centre if it is closest to the centre value.
    d. Update the min and max for this bin points

Data : 3, 13, 7, 5, 21, 23, 39, 23, 40, 23, 14, 12, 56, 23, 29

Sorted data : 3, 5, 7, 12, 13, 14, 21, 23, 23, 23, 23, 29, 39, 40, 56

Calculating five number summary : min, q1, q2, q3, max

0%	25%	50%	75%	100%
3.0	12.5	23.0	26.0	56.0

Determining the bins:
Range 1 : 3-23
Range 2 : 12.5-26
Range 3 : 23-56
Calculating Bin Density(MinD) : $\sqrt{n} = \sqrt{15} = 3.8729 \approx 3$

Data points	0%	25%	50%	75%	100%
3	0	9.5	20	23	53
5	2	7.5	18	21	51
7	4	5.5	16	19	49
12	9	0.5	11	14	44
13	10	0.5	10	13	43
14	11	1.5	9	12	42
21	18	8.5	2	5	35
23	20	10.5	0	3	33
23	20	10.5	0	3	33
23	20	10.5	0	3	33
23	20	10.5	0	3	33
29	26	16.5	6	3	27
39	36	26.5	16	13	17
40	37	27.5	17	14	16
56	53	43.5	33	30	0

Bin 1 : 12, 13, 14

Bin 2 : 21, 23, 23, 23, 23

iBin 3 : 39, 40

If Density of Bin 1 >= MinD(3) and Density of Bin 2 >= MinD(3) and Density of Bin 3 >= MinD(3) then all 3 center of the bins will be considered as valid centroids.
Therefore, Valid centers here are Centroid(12.5) and Centroid(23).
Now we remove the points from bins of validated centroids and repeat the same process again.
So now the new data will be 3, 5, 7, 29, 39, 40, 56.

Now again we will repeat this process.

Version 1.0

One of the problem with this algorithm was that there was no provision for operating on multi-dimensional data.
With increase in number of dimension from one to n there were n number of quartiles found but it was unknown how to consider a quartile for multi-dimensional data.
The solution to this problem was using the points given by combination of all the first, second and third quartiles of n dimensions.
For ex. a 3d data in dimension x, y and z, the first quartile Q1 will be the q1 from x, q1 from y and q1 from z.
Q1=[q1(x),q1(y),q1(z)]

Version 1.0 (Conclusion)

The centroids found in the form of quartiles could not qualify as valid centroids as they did not surpass the threshold of density i.e. that is minD = $\sqrt{n}$, therefore number of centroids found were not close enough or equal to ideal number of centers expected.
Also the sum of squared errors after applying the algorithm decreases to 85.2% from 94.6%

Version 1.1

Based on the outcomes of version 1.0, expected results were not observed, proper centroids in most of the data sets were not found and data containing centroids did not result in good clustering (conclusion based on the sum of square errors). So the necessity to make a change in algorithm arose. It was assumed that as less number of centres were obtained as compared to the ideal number of centroids found, change in minD may give improved results.

So algorithm was applied again changing the value of minD from $\sqrt{n}$ to $\sqrt{(n/2)}$.This change in density threshold was expected to increase the number of centroids found as now the centres with comparatively lesser no. of points could also validate and the sum of squarred errors might improve too.

Version 1.1 (Conclusion)

When results were checked after changing the minD value, enough points were still not obtained as for the ideal number of centroids that shoud be found, nor were the found centroids accurate enough (conclusion based on the sum of square errors) Therefore the density threshold needed to be further decreased from mind = $\sqrt{(n/2)}$ to $\sqrt{(n)/2}$

Version 1.2

With conclusion from 1.1, in order to get more validated centres, minD = $\sqrt{(n)/2}$ was taken into consideration. This change in density threshold was expected to increase the number of centroids found as now the centers with even lesser no. of points could validate and the sum of squarred errors might improve.

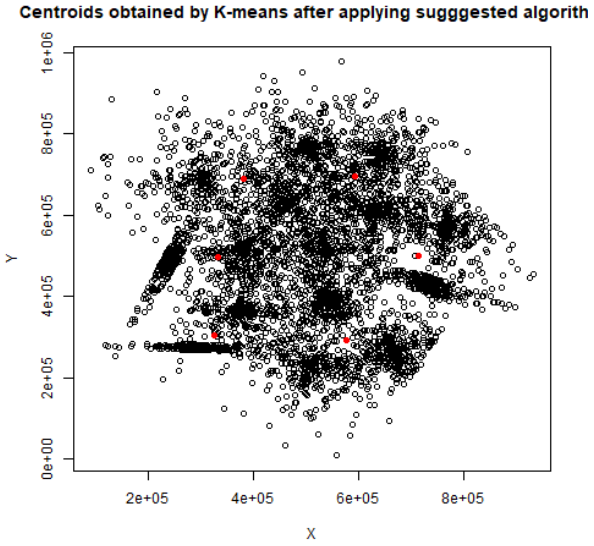
Version 1.2 (Conclusion)

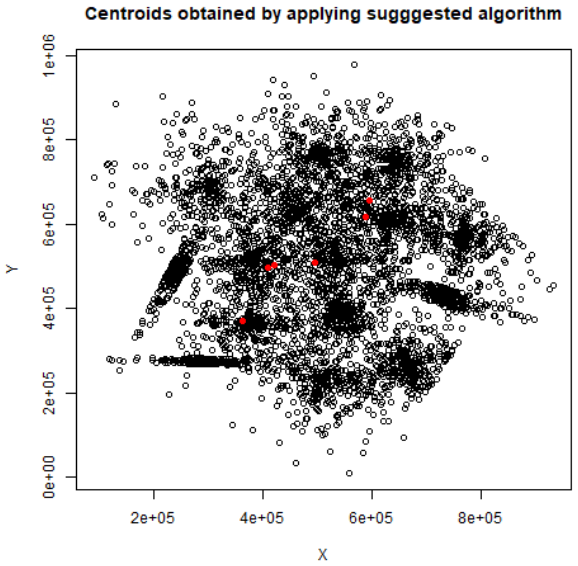
Not enough points as compared to ideal set of centroids were found and no change was seen in the sum of squared errors from the previous version.

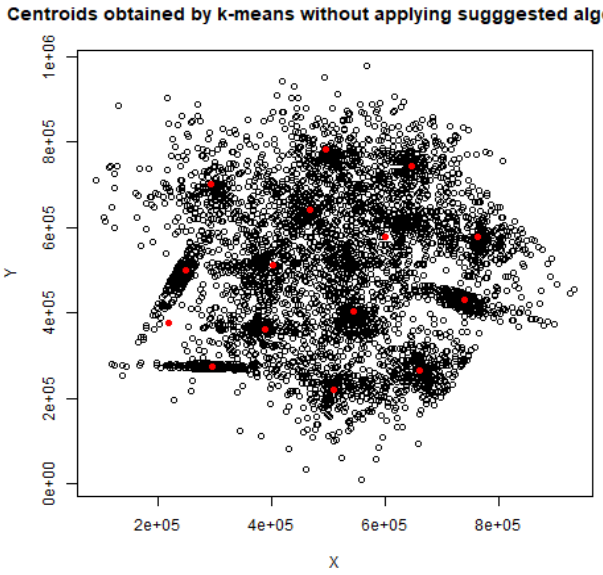
Data – S4

minD	Cluster provided	Cluster obtain by algorithm	Sum of squared Errors (%)	Time taken by algorithm	Time taken by k-means after applying algorithm	Time taken by k-means	Sum of squared errors (%) Of k-means
(√n) / 2	15	7	85.2	48.45	0.028	0.037	94.6
(√n)	15	6	81.5	40.637	0.0179	0.065	94.1
(√n/2)	15	6	81.5	43.495	0.0229	0.086	94.1

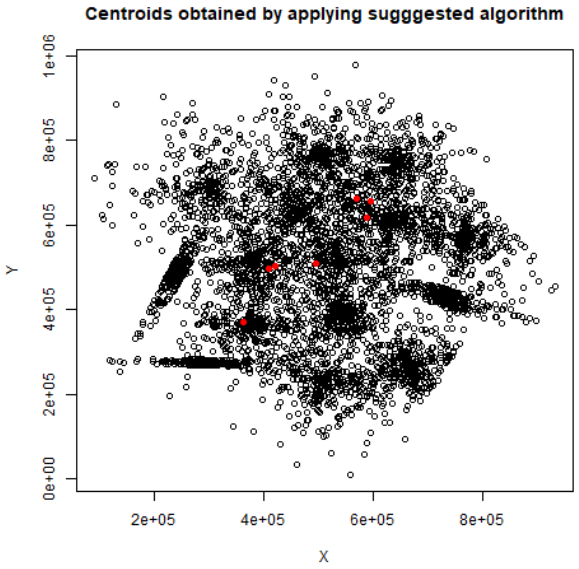
MinD – (√n)

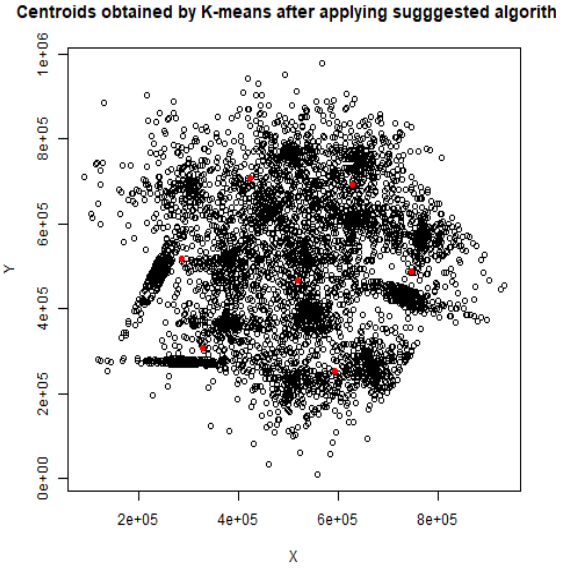

Centroids obtained by K-means after applying sugggested algorith
Y
X
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

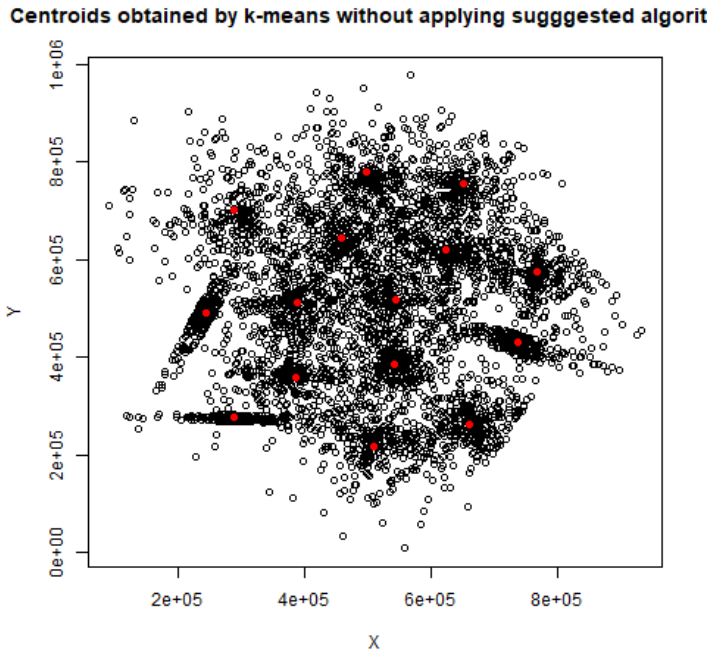

Centroids obtained by applying sugggested algorithm
Y
X
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06


Centroids obtained by k-means without applying sugggested algorit
Y
X
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

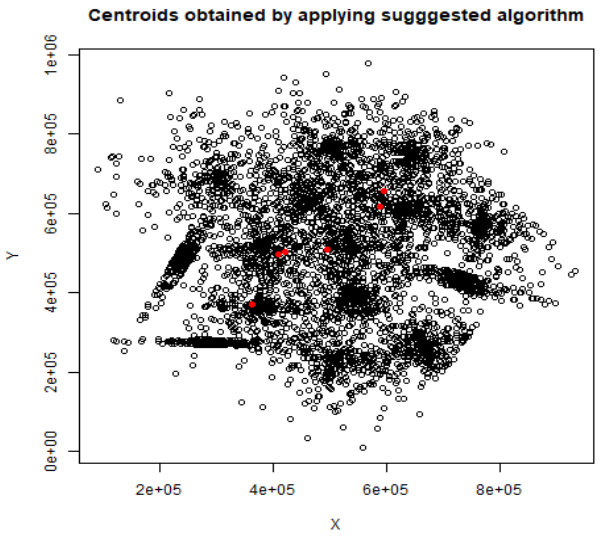
MinD – (√n)/2

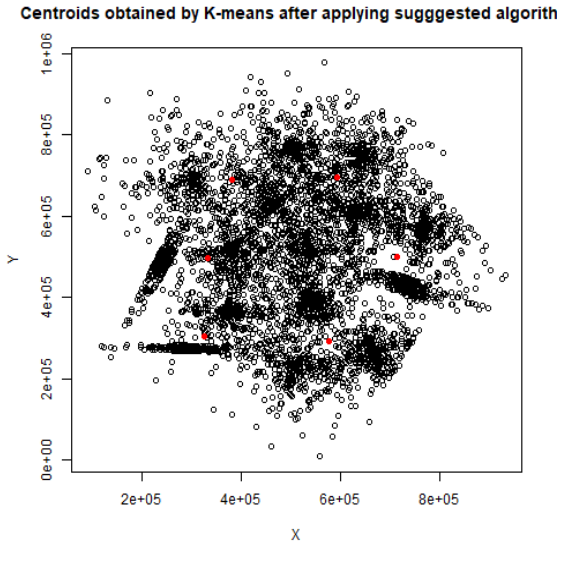

Centroids obtained by applying sugggested algorithm
Y
X
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06

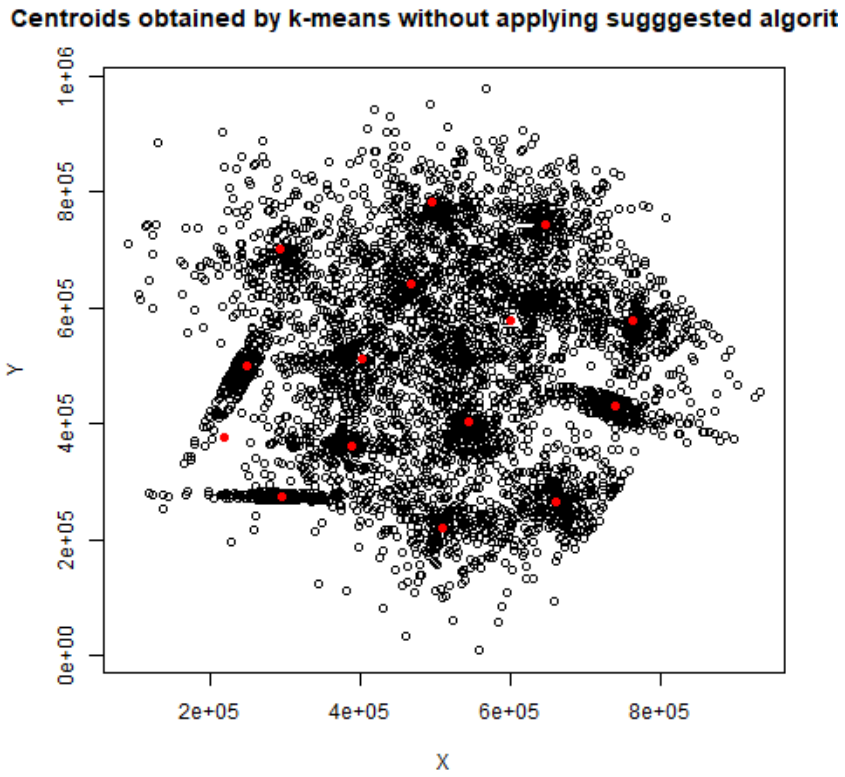

Centroids obtained by K-means after applying sugggested algorith
Y
X
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06


Centroids obtained by k-means without applying sugggested algorit
Y
X
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06
2e+05
4e+05
6e+05
8e+05

MinD – (√n/2)

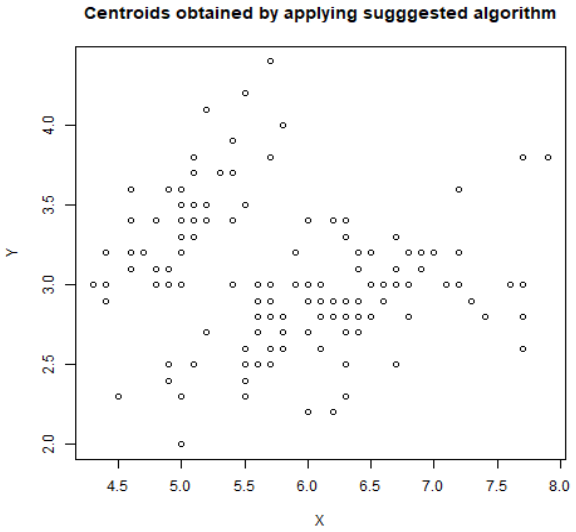

Centroids obtained by applying sugggested algorithm
Y
X
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06
2e+05
4e+05
6e+05
8e+05

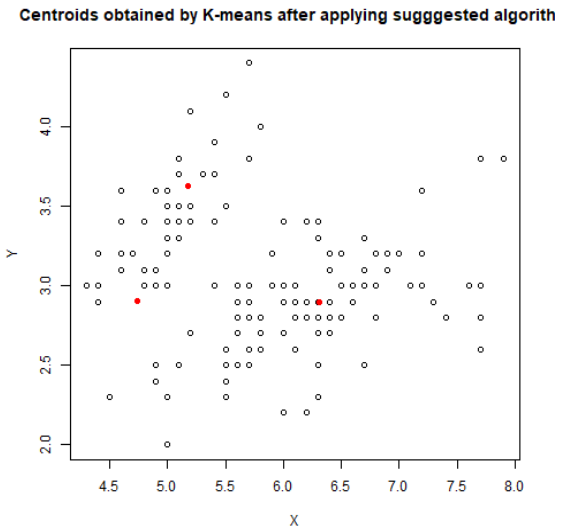

Centroids obtained by K-means after applying sugggested algorith
Y
X
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06
2e+05
4e+05
6e+05
8e+05

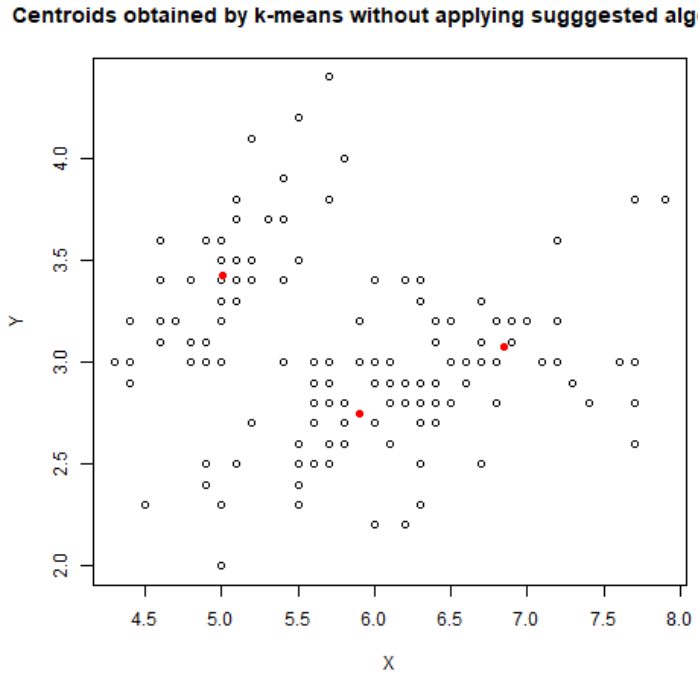


Data – Iris

minD	Cluster provided	Cluster obtain by algorithm	Sum of squared Errors (%)	Time taken by algorithm	Time taken by k-means after applying algorithm	Time taken by k-means	Sum of squared errors (%) Of k-means
(√n) / 2	3	None	None	None	None	None	None
(√n)	3	None	None	None	None	None	None
(√n/2)	3	None	None	None	None	None	None







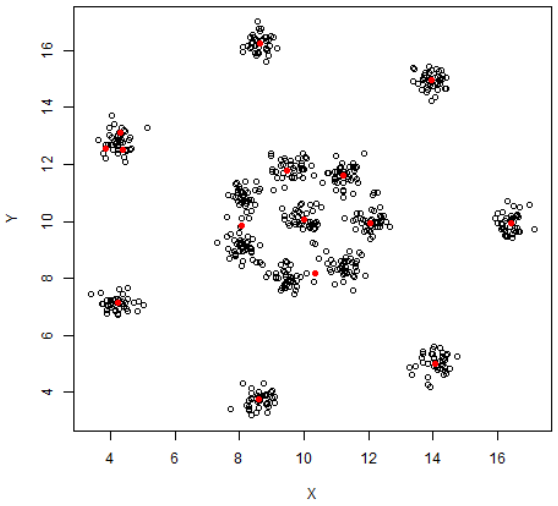
Data – R15

minD	Cluster provided	Cluster obtain by algorithm	Sum of squared Errors (%)	Time taken by algorithm	Time taken by k-means after applying algorithm	Time taken by k-means	Sum of squared errors (%) Of k-means
(√n) / 2	15	5	75.6	36.933	0.0039	0.0019	97.7
(√n)	15	5	75.6	48.629	0.0017	0.0019	98.2

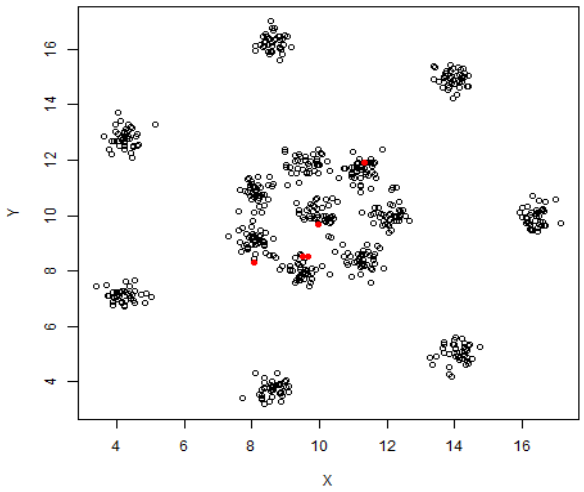
(√n/2)	15						

MinD – (√n)

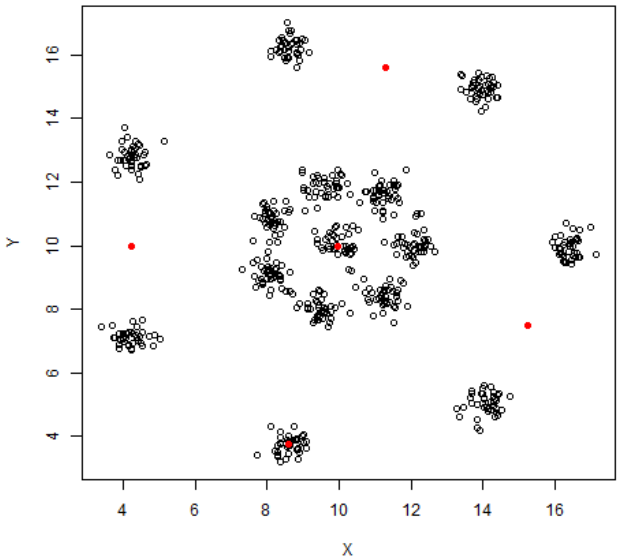
Centroids obtained by k-means without applying sugggested algorit



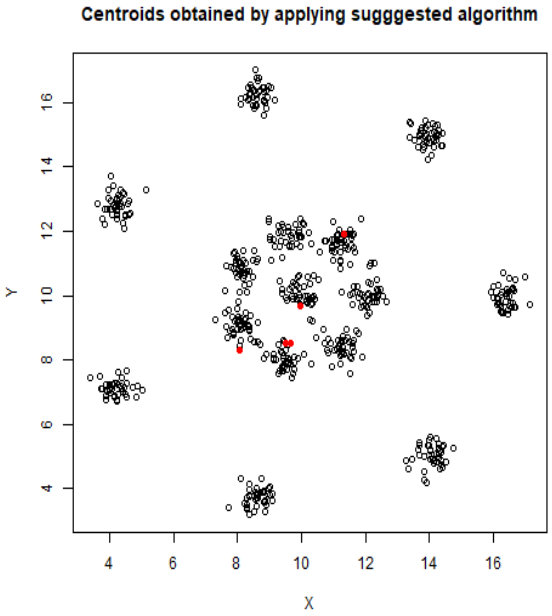
Centroids obtained by applying sugggested algorithm

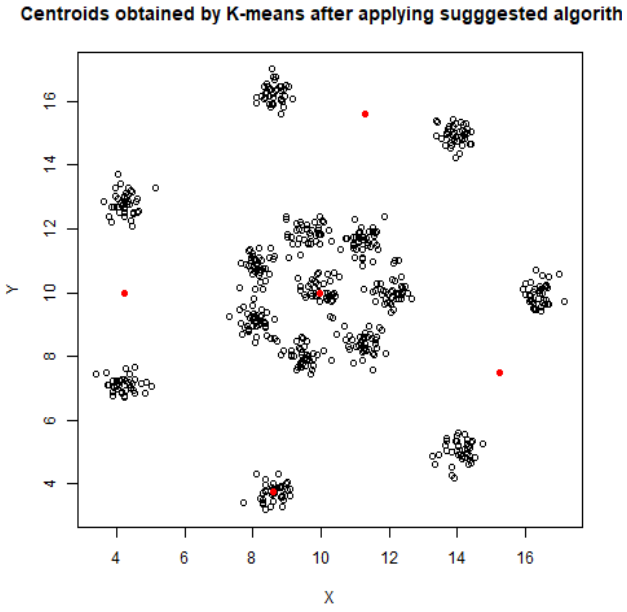


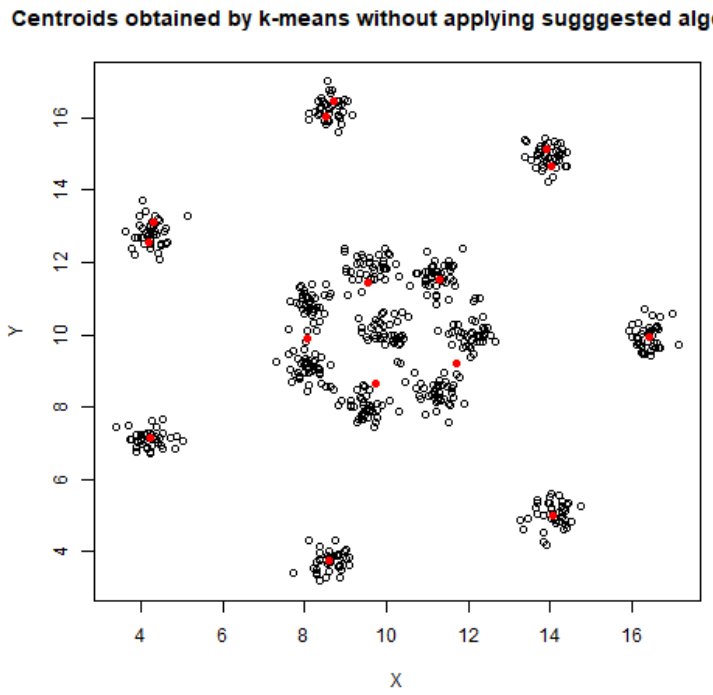
Centroids obtained by K-means after applying sugggested algorith



MinD – (√n) /2


Centroids obtained by applying sugggested algorithm
Y
X
4
6
8
10
12
14
16


Centroids obtained by K-means after applying sugggested algorith
Y
X
4
6
8
10
12
14
16



Dim032

minD	Cluster provided	Cluster obtain by algorithm	Sum of squared Errors (%)	Time taken by algorithm	Time taken by k-means after applying algorithm	Time taken by k-means	Sum of squared errors (%) Of k-means
(√n) / 2	16	None	None	None	None	None	None
(√n)	16	None	None	None	None	None	None
(√n/2)	16	None	None	None	None	None	None

Version 2.0

With respect to conclusion from 1.2,under the assumption that the quartiles of multi-dimensional data might not be accurate, the process to find quartiles was changed to following:

Find distance of each point from origin using Euclidian distance, and then find Quartiles of those distances.

Then take the nearest point of that quartile Distance from original data and Consider that point new as Quartile and continue the same process after finding the quartiles.

Conclusion (2.0)

The number of centroids were still not close enough to the expected number of centroids ideally, neither was there any improvement in the sum of squarred errors.

Version 2.1

The major drawback of version 2.0 was that the number of centroids it gave were still not equal to the ideal number of centers that should be found and furthermore, no improvement was seen in the sum of squared errors.

The need for more number of centroids is increasing with each verion in progression, so then it was decided that if a centre has atleast one point other than itself closer than the upper and lower limit of the range of values, then that is a valid centroid.

So now, minD = 1 and remaining procedure was kept same as version 2.0

Once all the centroids are found, they will be merged

This merger will be:

1. Find the distance of each centroid from every other centroid
2. Find the mean and standard deviations of these distances
3. Considering confidence level to be 90% , Confidence Interval ci = mean – 1.65(standard deviation)
4. Any 1 of those centroid is selected, and if any other centroid is found to be at a distance less than ci, the mean of the 2 centroids is chosen and replaced in the centroids.
5. Again the distances of each centroid (including new and removing those 2) from every other centroid is calculated, but now the ci will stay as it was

Conclusion (Version 2.1)

The number of centroids were now much more, and were to be merged with those closer to each other by a distance CI or less, but in a few datasets, the CI turned out to be a negative value and none of the centroids merged, which is an unexpected outcome.So the conclusion is that the formula to find confidence interval needs to change.

Conclusion

The goal of this algorithm was to help k-means algorithm reach to the accurate centers faster, which is still under progress and is being worked on. Various methods to achieve the goal have been tried till now, but not so fruitful yet.Conclusion can be drawn from the procedures applied till now, that finding a correct way to calculate measures of central tendency is a tricky thing, and the centroids to be found need to be accurate and close enough or equal to the required number of centers ideally.