Partitioning and hierarchical clustering

BHUVANESH JEEVARATHINAM

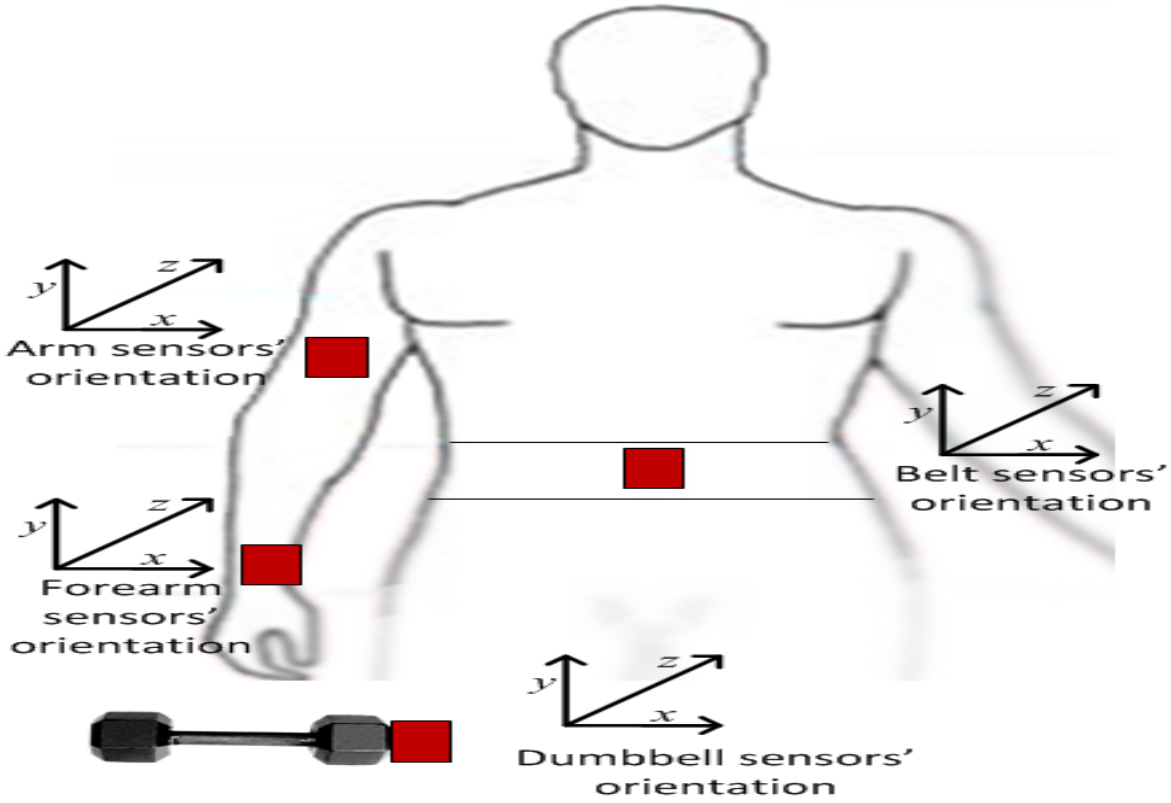
STA-6704

Partitioning and hierarchical clustering

Problem Solution:

The main objective of this project is to cluster the human activity based on different readings of the sensors attached at different parts of the body. We tend to group the activity which has similar orientation readings.

The data is measured from Accelerometer sensor which is attached in 4 different body parts to measure various orientations. These accelerometers is attached to a subject based on his weight lifting activity to monitor his activity.



The following symbol " ■ " designates one of the set of sensors described in the text

Dataset Description:

The data set has following features.

user (text)

gender (text)

age (integer)

how_tall_in_meters (real)

weight (int)

body_mass_index (real)

x1 (type int, contains the read value of the axis 'x' of the 1st accelerometer, mounted on waist)

y1 (type int, contains the read value of the axis 'y' of the 1st accelerometer, mounted on waist)

z1 (type int, contains the read value of the axis 'z' of the 1st accelerometer, mounted on waist)

x2 (type int, contains the read value of the axis 'x' of the 2nd accelerometer, mounted on the left thigh)

y2 (type int, contains the read value of the axis 'y' of the 2nd accelerometer, mounted on the left thigh)

z2 (type int, contains the read value of the axis 'z' of the 2nd accelerometer, mounted on the left thigh)

x3 (type int, contains the read value of the axis 'x' of the 3rd accelerometer, mounted on the right ankle)

y3 (type int, contains the read value of the axis 'y' of the 3rd accelerometer, mounted on the right ankle)

z3 (type int, contains the read value of the axis 'z' of the 3rd accelerometer, mounted on the right ankle)

x4 (type int, contains the read value of the axis 'x' of the 4th accelerometer, mounted on the right upper-arm)

y4 (type int, contains the read value of the axis 'y' of the 4th accelerometer, mounted on the right upper-arm)

z4 (type int, contains the read value of the axis 'z' of the 4th accelerometer, mounted on the right upper-arm)

Data Source:

http://groupware.les.inf.puc-rio.br/har#dataset

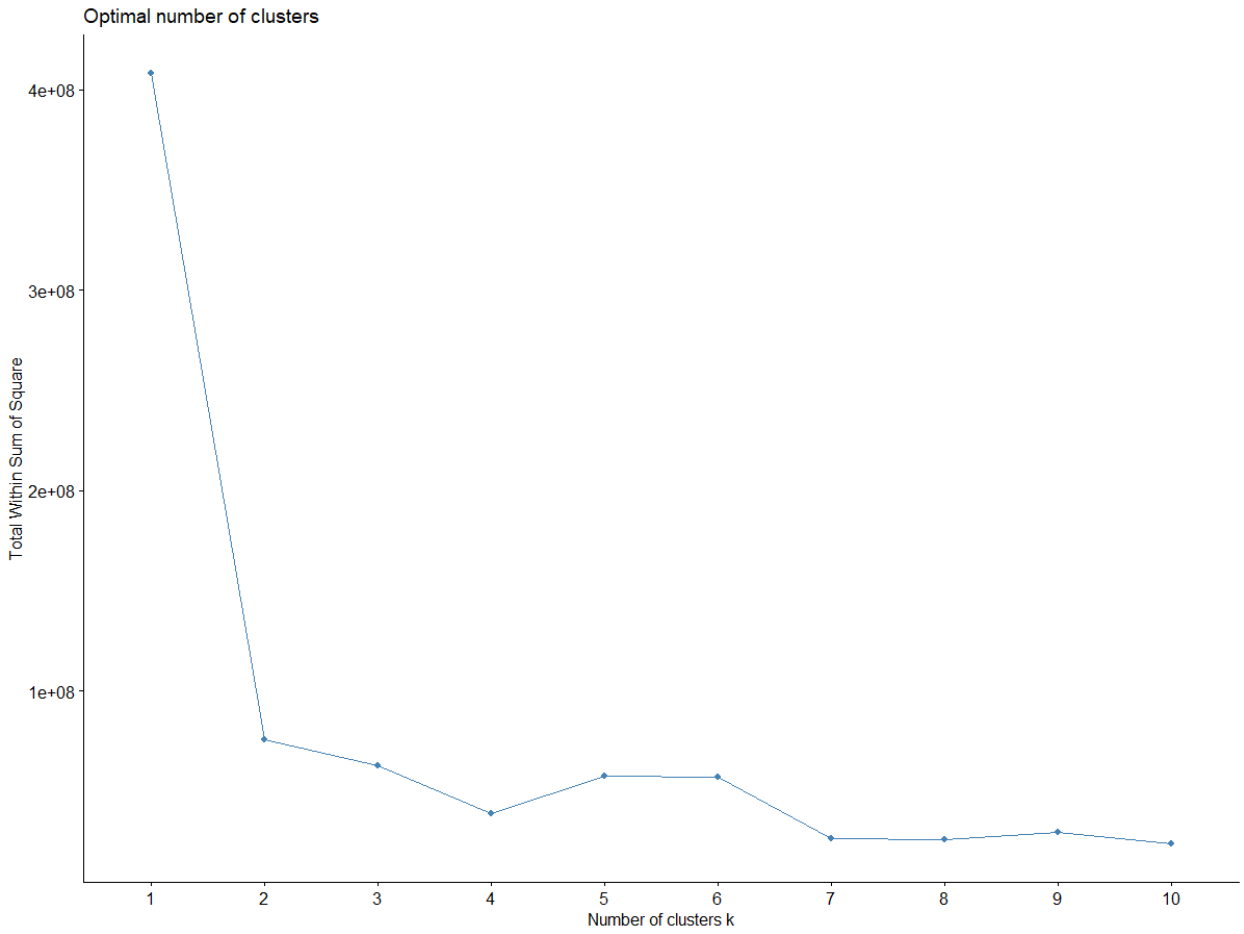
Data Exploration in R:

Initially data is loaded in R with delimited set as ";". Once the data is loaded, the data is checked for missing values. We found that there are no missing values in data. Only the axis features are sliced and taken to fit in the algorithm. We can neglect columns such as user name etc which may not be useful to determine the activity recognition.

Partitioning Clustering

Optimal Clusters:

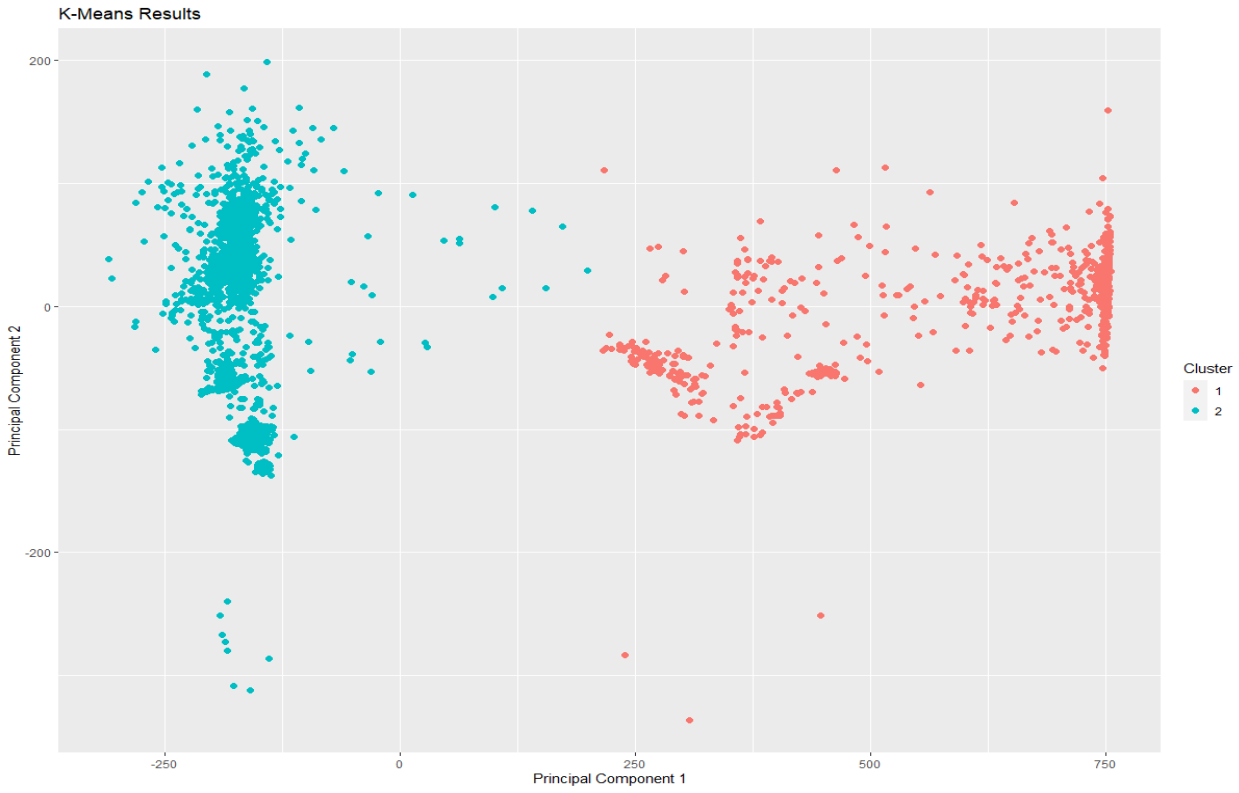
We use Elbow method to determine the optimal number of clusters. We tend to plot the curve with respect to wittiness distance and the number of clusters.



From this plot we can confirm that taking k =2 as the optimal number of clusters. This also proves our hypothesis to group based on standing or sitting activity as the initial phase of clustering as these two activities show a greater deviation in the measurements of the sensor readings.

K-means:

First, we choose K-means algorithm for partitioning.



Cluster stats:

Considering a few important stats alone,

$cluster.number
[1] 2

$cluster.size
[1] 2577 736

$min.cluster.size

[1] 736

$diameter
[1] 1283.660 1309.062

$average.distance
[1] 150.4088 283.8113

$median.distance
[1] 143.9201 260.7834

$average.between
[1] 763.1832

$average.within
[1] 180.0449

$n.between
[1] 1896672

$n.within
[1] 3589656

$max.diameter
[1] 1309.062

$min.separation
[1] 113.1503

$pearsongamma
[1] 0.8897197

$dunn
[1] 0.08643621

Clustering in Large Applications (CLARA):

CLARA Considers only a sample set from the population and applies the algorithm and generate a medoids.

CLARA STATS:

$average.between
[1] 367.0302

$average.within
[1] 373.3759

$n.between
[1] 1866960

$n.within
[1] 3619368

$max.diameter
[1] 1520.58

$min.separation
[1] 1

$within.cluster.ss
[1] 407751235

$pearsongamma
[1] -0.01416936

$dunn
[1] 0.0006576439

Fuzzy Analysis Clustering (FANNY):

Computes a fuzzy clustering of the data into k clusters.

FANNY STATS:

$diameter
[1] 1520.580 1458.153

$average.distance
[1] 377.5037 359.2858

$median.distance
[1] 197.4437 192.6473

$average.between
[1] 368.5156

$average.within
[1] 373.44

$n.between

[1] 1902186

$n.within
[1] 3584142

$max.diameter
[1] 1520.58

$min.separation
[1] 1

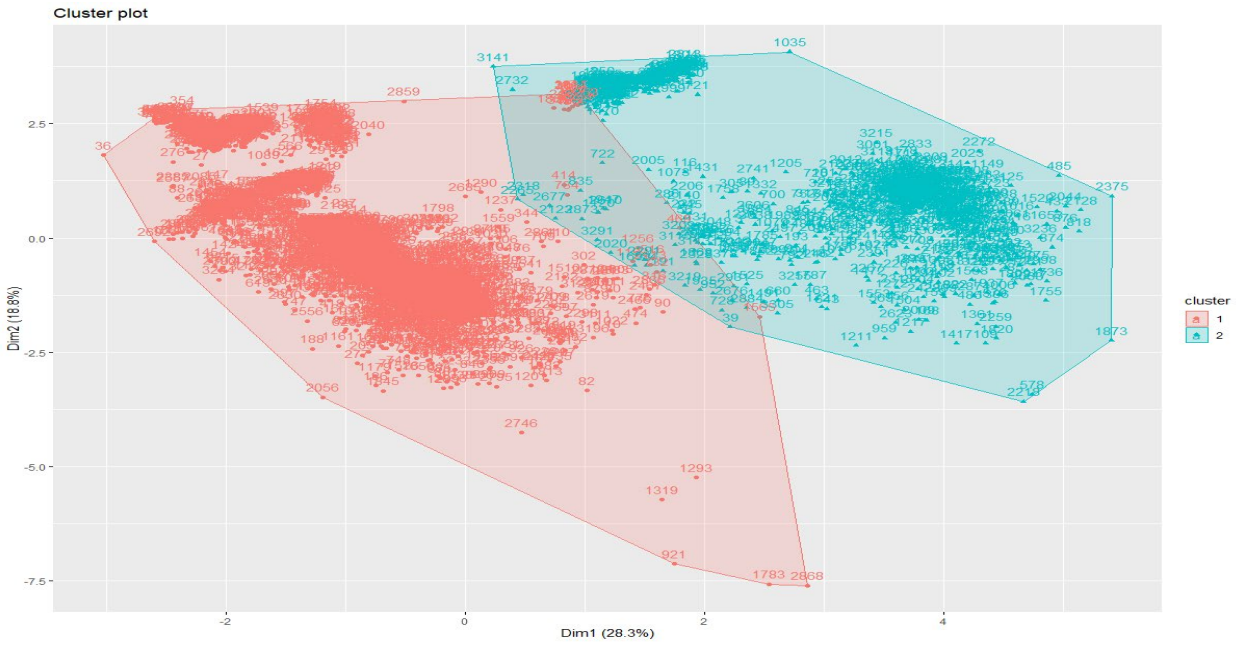
$within.cluster.ss
[1] 407856157

$pearsongamma
[1] -0.01106358

$dunn
[1] 0.0006576439

Partitioning Around Medoids (PAM):

Partitioning (clustering) of the data into k clusters “around medoids”, a more robust version of K-means.



$diameter
[1] 1520.580 1458.153

$average.distance
[1] 378.4941 354.9431

$median.distance
[1] 197.8863 191.1962

$average.between
[1] 367.0302

$average.within
[1] 373.3759

$n.between
[1] 1866960

$n.within
[1] 3619368

$max.diameter
[1] 1520.58

$within.cluster.ss
[1] 407751235

$pearsongamma
[1] -0.01416936

$dunn
[1] 0.0006576439

COMPARIOSN OF MODEL:

Based on the various stats of the functions we compare few important cluster terms.

- Higher the Dunn Index, greater is the clustering, Hence Kmeans has higher Dunn
  Kmeans (0.08641) > Clara >= FANNY >= PAM

- Visual Representation shows that the clusters are grouped well when there is no overlap between them. This also confirms that K-means algorithm more effectively clusters as from the plot we can see clear clusters without or negligible overlapping.

- Higher values of Pearson Gamma indicate that the clustering is done well.
  K-means (0.8897197) > Clara >= Pam > fanny

- Average Within Difference signifies the average distance measure between the clusters. The more the distance the diverse the clusters and overlaps with other clusters.
  K-means(180.449) < CLARA (373.3) < PAM (373.34) < Fanny (373.37)

BEST MODEL:

From the above inference and observation, we can conclude that optimal number of clusters is 2 for the given unsupervised task and when Kmeans Algorithm is used the clustering of human activity is highly efficient and distinct for efficient recognition.

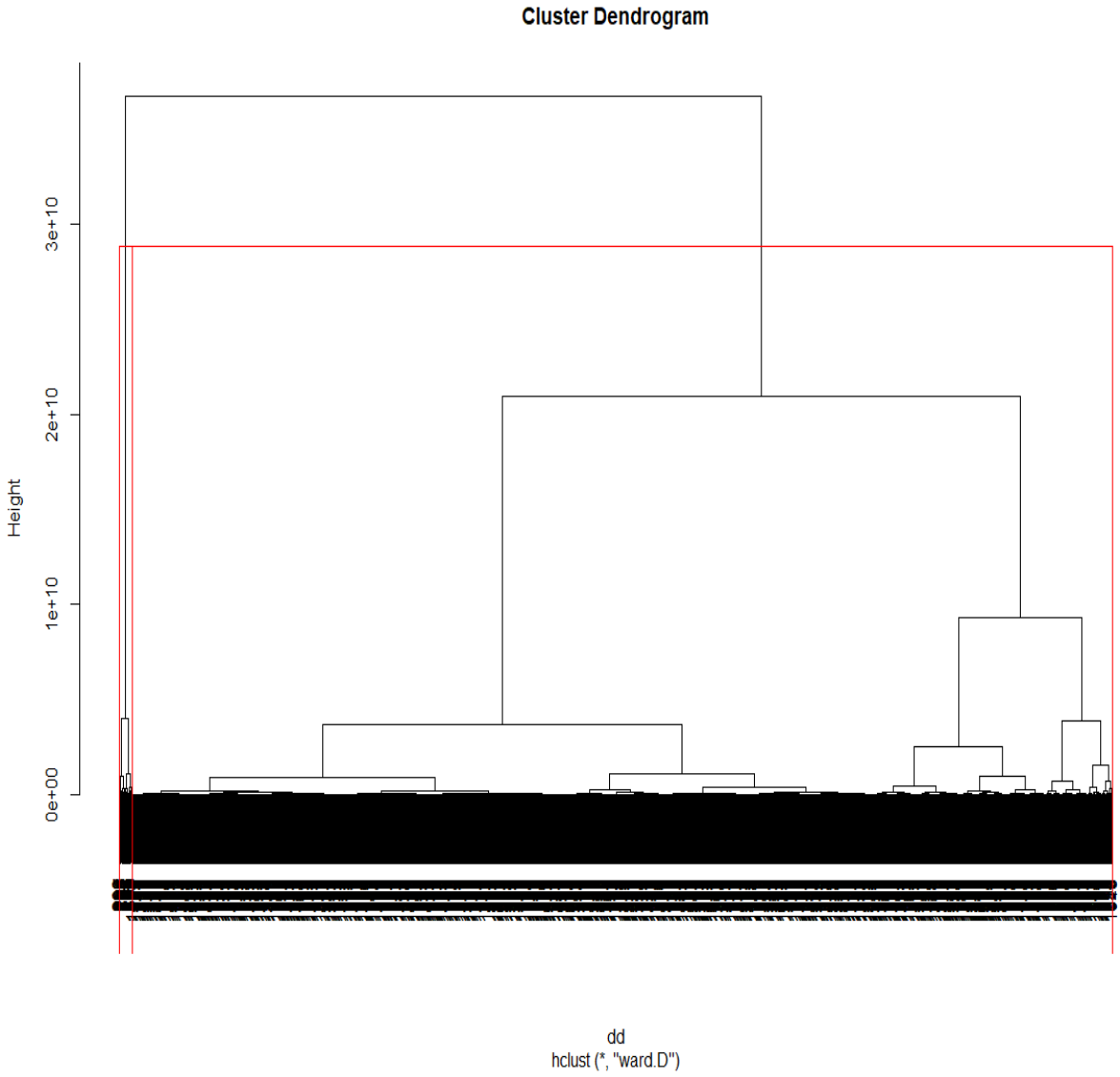
HIERARCHIAL CLUSTERING

With the given dataset we tend to cluster the groups based on various hierarchical functions to ensure the better grouping in achieved.

HCLUST:

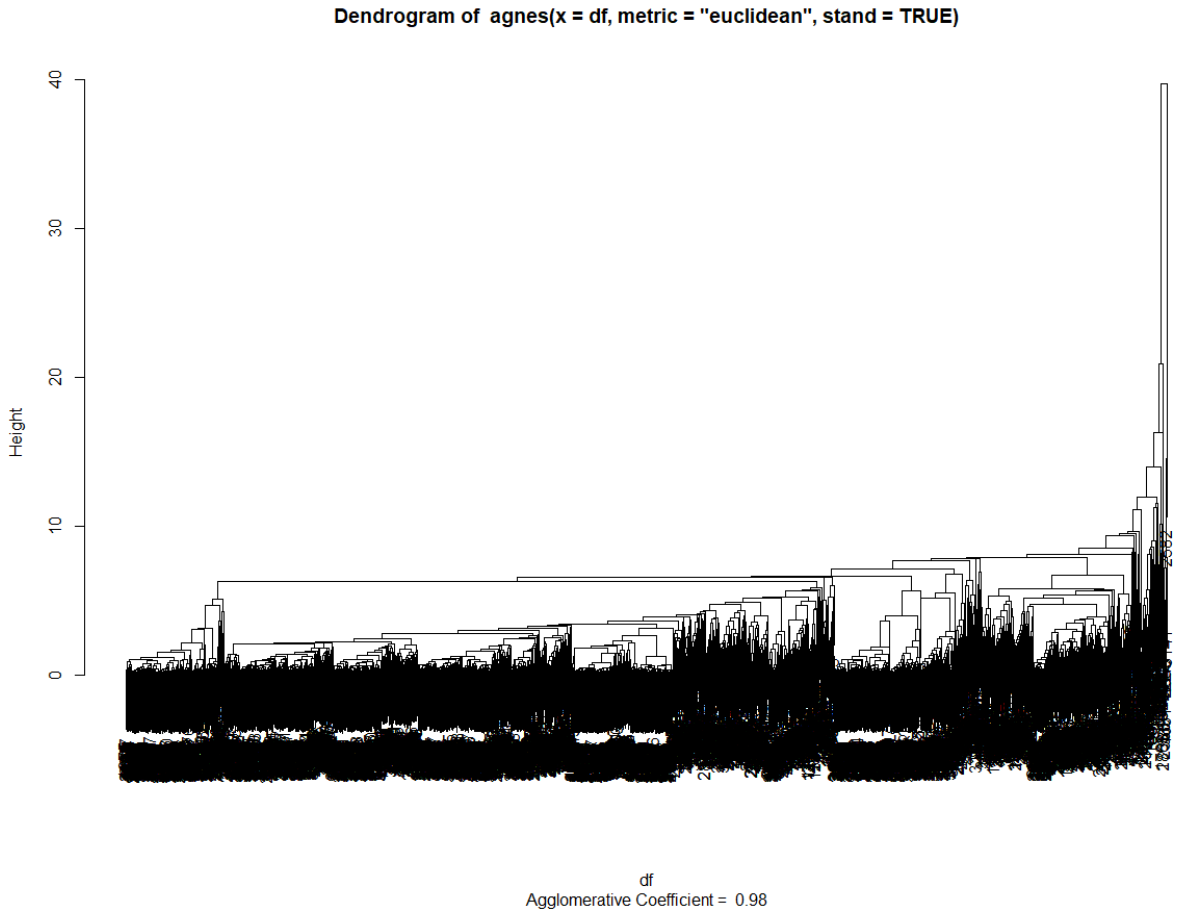
In this Clustering we tend to draw a threshold on the dendrogram which has highest horizontal distance. We can conclude that there are 2 optimal clusters.

In Hyper parameter for method we use “Ward” which uses minimum variance method for hierarchical clustering.



This dendrogram, clearly helps us visualize the clusters which are formed based on the dataset.as per this use case we tend to form clusters based on two important activity associated with sitting and standing.

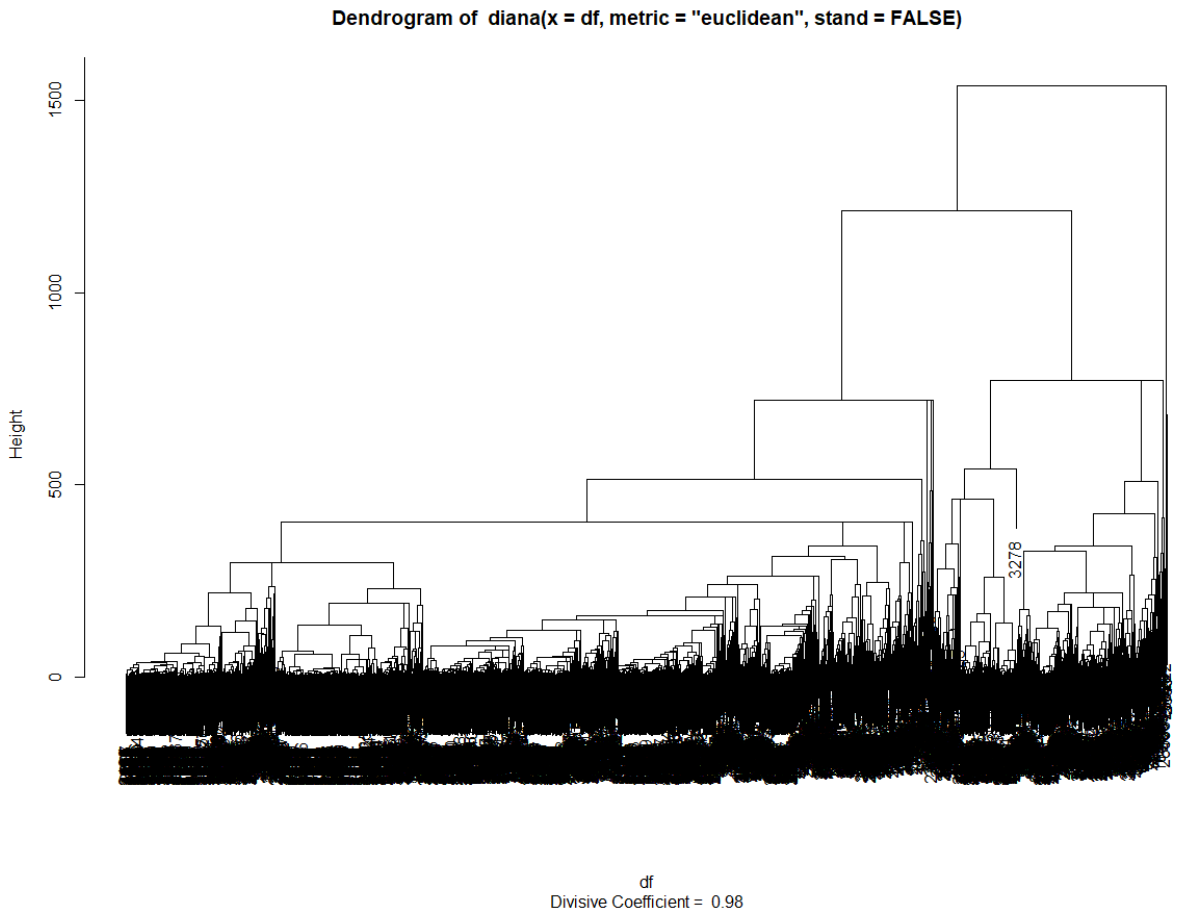
Agglomerative Nesting (AGNES)



Agnes works in Bottom -Up manner. It is usually good in identifying small clusters. At each step of the algorithm, the two clusters that are the most similar are combined into a new bigger cluster (nodes). This procedure is iterated until all points are member of just one single big cluster (root) (see figure below).

Agglomerative coefficient is 0.98. It generally ranges from 0 to 1. Whichever close to 1 is good in clustering. Hence this does fairly good job.

DIVISIVE ANALYSIS CLUSTERING (DIANA):



Diana generally works from top- bottom approach. It is inverse order of AGNES. Divisive is generally good in detecting large clusters. At each step of iteration, the most heterogeneous cluster is divided into two. The process is iterated until all objects are in their own cluster

The divisive coefficient is calculated as 0.98. This infers that the clustering is similar to AGNES.

Other Functions like MONA cannot be used at it is only used for binary variables., which does not suit for this dataset.

MODEL COMPARISON AND CHOOSING BEST MODEL:

From the observations above, we can conclude few important inferences.

- HCLUST is the algorithm available in stats package, whereas AGNES and DIANA are available from Cluster package.
- AGNES and DIANA Works in Inversive way. One follows Bottom up Approach and other vice-versa.
- The coefficient of both remains equal for this particular case, Hence there is no much difference. However, since Divisive operates from Top-Bottom and is easy to cluster larger groups, in this scenarios DIANA performs better.
- Generally, overall HCLUST is chosen as the clusters formed are clear and more differentiated

RESULT:

Unsupervised Learning helped in forming clusters which effectively grouped activity based on various reading from the accelerometer sensor. We observed that the activities can be broadly clustered into two types namely ***Sitting*** and ***Standing***.

CodeLink:

https://github.com/bhuvaneshkj/Clustering-Unsupervised-