ASSIGNMENT 1

CLUSTERING TECHNIQUES ON STUDENT KNOWLEDGE DATASET

This assignment provided us with a dataset of StudentKnowledge in which the Clustering techniques were to be applied to know the number of clusters to be necessary if it were to be made into a classification problem. (Basically, the number of inference that can be obtained from the dataset to convert it to a classification problem)

A quick summary of the dataset indicates that:

* It has no NA values.
* It has 402 observations.
* It depends on 5 features/dimensions.

Kmeans Clustering:

In Kmeans clustering, since we don’t have a pre-conceived idea about the dataset, we try to use the “NbClust” package which helps us in determining which can be the best number of clusters that can be used for clustering the dataset.

The dataset is scaled and converted to a data frame(if it isn’t one initially). Once the dataset is scaled, we have the method “kmeans” to be selected in “NbClust” package and try to find the best cluster by:

|  |
| --- |
| nc <- NbClust(scale(StudentKnowledgeData), min.nc=2, max.nc=15, method="kmeans")  \*\*\* : The Hubert index is a graphical method of determining the number of clusters.  In the plot of Hubert index, we seek a significant knee that corresponds to a  significant increase of the value of the measure i.e the significant peak in Hubert  index second differences plot.    \*\*\* : The D index is a graphical method of determining the number of clusters.  In the plot of D index, we seek a significant knee (the significant peak in Dindex  second differences plot) that corresponds to a significant increase of the value of  the measure.    \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Among all indices:  \* 5 proposed 2 as the best number of clusters  \* 4 proposed 3 as the best number of clusters  \* 1 proposed 4 as the best number of clusters  \* 2 proposed 6 as the best number of clusters  \* 1 proposed 7 as the best number of clusters  \* 2 proposed 8 as the best number of clusters  \* 1 proposed 12 as the best number of clusters  \* 2 proposed 13 as the best number of clusters  \* 3 proposed 14 as the best number of clusters  \* 2 proposed 15 as the best number of clusters  \*\*\*\*\* Conclusion \*\*\*\*\*    \* According to the majority rule, the best number of clusters is 2      \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |

Thus, using the NbClust method we discovered that the best cluster for the dataset will be 2.

This indicates that we can have a KnowledgeLevel classifier to have two classes “A” and “B” which would be assigned to the 402 observations (any one of the two classes) to make it to a classification problem. The accuracy/results of the Kmeans clustering to a particular cluster/class cannot be identified in this dataset because we don’t have an output class which is assigned to the Knowledgedataset that is provided to us.

Hierarchical Clustering:

In hierarchical clustering for the given dataset, the assumption is to have “n” clusters, in this case it is 402 clusters and after which the clusters are reduced/fused based on the dissimilarity between the points/clusters.

Different linkage methods for the Hierarchical clustering is tried and from my inference, Complete and average methods seems to closely be similar to the Kmeans clustering mechanism (which predicted 2 clusters to provide the best). Here since we don’t need to assume the clusters, we try to “cutree” based on the clusters that we want to have and obtain the results. So, based on the previous Kmeans clustering results, I tried to do the cutree method for 2 clusters and obtained the results for “complete”, “average” and “single” linkage methods and the results are as follows :

|  |
| --- |
| > **table(cutree(h\_clust.complete,2), myData$PEG)**    0 0.01 0.03 0.04 0.05 0.07 0.08 0.09 0.1 0.11 0.12 0.13 0.14 0.15 0.16 0.17 0.18 0.19 0.2 0.21 0.22 0.23 0.24  1 1 1 2 2 4 3 3 3 14 3 3 4 2 4 2 1 3 2 3 2 1 3 8  2 0 2 0 0 0 2 0 1 2 1 0 1 1 1 1 0 0 1 2 2 3 1 3    0.25 0.26 0.27 0.28 0.29 0.3 0.31 0.32 0.33 0.333 0.34 0.35 0.37 0.38 0.4 0.45 0.47 0.48 0.49 0.5 0.51 0.52 0.53  1 10 6 3 4 6 19 7 18 4 1 3 1 1 1 3 1 0 1 1 2 2 4 1  2 4 2 0 5 3 0 1 0 1 0 1 0 0 0 0 3 1 0 0 4 2 0 1    0.54 0.55 0.56 0.57 0.58 0.59 0.6 0.61 0.62 0.63 0.64 0.65 0.66 0.67 0.68 0.69 0.7 0.71 0.74 0.75 0.76 0.77 0.78  1 3 2 6 3 5 5 4 2 4 0 3 3 8 6 1 0 0 2 0 1 0 4 4  2 1 3 2 2 2 3 4 1 3 2 1 2 7 1 2 1 1 0 1 1 1 1 2    0.79 0.8 0.81 0.82 0.83 0.84 0.85 0.86 0.87 0.88 0.89 0.9 0.92 0.93 0.94 0.95 0.96 0.97 0.98 0.99  1 2 6 3 2 5 2 5 1 1 2 2 5 1 0 1 1 1 0 1 0  2 2 2 1 1 2 1 2 2 0 2 3 1 1 2 0 1 0 1 1 1 |
| > **table(fit.km$cluster,myData$PEG)**    0 0.01 0.03 0.04 0.05 0.07 0.08 0.09 0.1 0.11 0.12 0.13 0.14 0.15 0.16 0.17 0.18 0.19 0.2 0.21 0.22 0.23 0.24  1 1 3 2 2 4 5 3 4 16 4 3 5 3 5 3 1 3 3 5 4 4 4 10  2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1    0.25 0.26 0.27 0.28 0.29 0.3 0.31 0.32 0.33 0.333 0.34 0.35 0.37 0.38 0.4 0.45 0.47 0.48 0.49 0.5 0.51 0.52 0.53  1 14 7 3 8 8 19 8 16 5 1 4 1 1 1 3 3 0 0 0 3 1 2 1  2 0 1 0 1 1 0 0 2 0 0 0 0 0 0 0 1 1 1 1 3 3 2 1    0.54 0.55 0.56 0.57 0.58 0.59 0.6 0.61 0.62 0.63 0.64 0.65 0.66 0.67 0.68 0.69 0.7 0.71 0.74 0.75 0.76 0.77 0.78  1 1 1 4 2 3 0 1 0 0 0 0 1 2 0 0 0 0 0 0 0 0 0 0  2 3 4 4 3 4 8 7 3 7 2 4 4 13 7 3 1 1 2 1 2 1 5 6    0.79 0.8 0.81 0.82 0.83 0.84 0.85 0.86 0.87 0.88 0.89 0.9 0.92 0.93 0.94 0.95 0.96 0.97 0.98 0.99  1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  2 4 8 4 3 7 3 7 3 1 4 5 6 2 2 1 2 1 1 2 1 |
| > **table(cutree(h\_clust.average,2), myData$PEG)**    0 0.01 0.03 0.04 0.05 0.07 0.08 0.09 0.1 0.11 0.12 0.13 0.14 0.15 0.16 0.17 0.18 0.19 0.2 0.21 0.22 0.23 0.24  1 1 3 2 2 4 5 3 4 16 4 3 5 3 5 3 1 3 3 5 4 4 4 9  2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2    0.25 0.26 0.27 0.28 0.29 0.3 0.31 0.32 0.33 0.333 0.34 0.35 0.37 0.38 0.4 0.45 0.47 0.48 0.49 0.5 0.51 0.52 0.53  1 14 7 3 7 8 19 8 18 5 1 4 1 1 1 3 4 0 1 1 5 4 4 2  2 0 1 0 2 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0    0.54 0.55 0.56 0.57 0.58 0.59 0.6 0.61 0.62 0.63 0.64 0.65 0.66 0.67 0.68 0.69 0.7 0.71 0.74 0.75 0.76 0.77 0.78  1 4 5 8 4 6 8 8 3 7 2 4 4 15 6 3 1 0 2 0 2 1 5 6  2 0 0 0 1 1 0 0 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0    0.79 0.8 0.81 0.82 0.83 0.84 0.85 0.86 0.87 0.88 0.89 0.9 0.92 0.93 0.94 0.95 0.96 0.97 0.98 0.99  1 4 8 4 3 7 3 5 3 1 3 4 6 2 2 1 1 1 1 2 1  2 0 0 0 0 0 0 2 0 0 1 1 0 0 0 0 1 0 0 0 0 |
| > **table(cutree(h\_clust.single,2), myData$PEG)**    0 0.01 0.03 0.04 0.05 0.07 0.08 0.09 0.1 0.11 0.12 0.13 0.14 0.15 0.16 0.17 0.18 0.19 0.2 0.21 0.22 0.23 0.24  1 1 3 2 2 4 5 3 4 16 4 3 5 3 5 3 1 3 3 5 4 4 4 11  2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0    0.25 0.26 0.27 0.28 0.29 0.3 0.31 0.32 0.33 0.333 0.34 0.35 0.37 0.38 0.4 0.45 0.47 0.48 0.49 0.5 0.51 0.52 0.53  1 14 8 3 9 9 19 8 18 5 1 4 1 1 1 3 4 1 1 1 6 4 4 2  2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0    0.54 0.55 0.56 0.57 0.58 0.59 0.6 0.61 0.62 0.63 0.64 0.65 0.66 0.67 0.68 0.69 0.7 0.71 0.74 0.75 0.76 0.77 0.78  1 4 5 8 5 7 8 8 3 7 2 4 5 15 7 3 1 1 2 1 2 1 5 6  2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0    0.79 0.8 0.81 0.82 0.83 0.84 0.85 0.86 0.87 0.88 0.89 0.9 0.92 0.93 0.94 0.95 0.96 0.97 0.98 0.99  1 4 8 4 3 7 3 7 3 1 4 4 6 2 2 1 2 1 1 2 1  2 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 |