Lab 3 Association Analysis II

Akshay Gurudath, Keshav Padiyar Manuru

11 March 2021

First, cluster the data with different algorithms and number of clusters. Use the Clusters to class evaluation model to see whether the clustering algorithm is able to discover the class division existing in the data

To start with we used simple K-Means clustering algorithm with number of Clusters = 2 since there were 2 classes in the given dataset.
However, the dataset is composed of discrete values. But K-means algorithm is applicable only when cluster mean or centroid is defined. That means it works only on the numerical data and not on the categorical data. Here, clustering is formed with the Euclidean distance of discrete variable and these clusters are not best ones.
We then used density based and Hierarchical clustering with number of clusters = 2. Below are the results of the same. We noticed that, when used Hierarchical clustering with Average linktype, the Mis-Clustering error was slightly less compared to alternatives. Generally, we noticed high mis-clustering error for all these clustering algorithms. We believe this is because of the distance metric and choosing the right distance metric for discrete data

Using Simple K-Means:

#	**Algorithm**	**SSE**	**MisClustering %**
1	Simple K-Means With K =2	358	47.58%
2	Simple K-Means With K =4	293	61.29%
3	Density Based Clustering On K-Means: K=2; minDev = 0.05	358	45.96%

Using Hierarchical Clustering; Number of Clusters = 2:

#	Link Type	MisClustering %
1	Single	49.19%
2	Complete	48.38%
3	Average	43.54%
4	Mean	46.77%
5	Centroid	48.38%

Association Analysis:
Try to find as few rules predicting class 1 as possible, i.e. try to remove redundant rules

Using Apriori algorithm with the specifications minSupport = 0.05 and maxNumber of rules = 19, we generated 19 rules. Out of 19 rules we removed some redundant rules and chose the best rules which can discriminate between class 1 and class 0.

Best Rules
1. attribute5=1 29 == class=1 29 conf:(1)
2. attribute1=3 attribute2=3 17 == class=1 17 conf:(1)
3. attribute1=2 attribute2=2 15 == class=1 15 conf:(1)
4. attribute1=1 attribute2=1 9 == class=1 9 conf:(1)

All the rules mentioned above represents the full population of class 1 (62 rows). We can also simplify the rules as cluster= 1 when attribute1=attribute2 or attribute5=1.

Approach to remove redundant rules:

These rules were generated along with the rule : **attribute5=1 29 == class=1 29 conf:(1)**

1. attribute5=1 attribute6=1 16 == class=1 16 conf:(1)
2. attribute5=1 attribute6=2 13 == class=1 13 conf:(1)

We can clearly notice that **attribute 6** is not required when **attribute 5** is already present in the rule as attribute 5 alone could discriminated major portion of the class 1.
In such way we identify other redundant rule and eliminate them.

Try to answer the question above. Finally, would you say that the clustering algorithms fail or perform poorly for the monk1 dataset? Why or why not?

We see that the rule attribute1=attribute2 or attribute5=1 is difficult to identify with the conventional algorithms using conventional distance metrics as the given dataset is composed of discrete observations. Therefore, these clustering algorithms performs poorly for the monk1 dataset.