LAB2 Association Rules

Wuhao Wang(wuhwa469) Hong Zhang(honzh073)

2/27/2022

Situation 1:

Algorithm: SimpleKmeans with 3 clusters Number of bins: 3

Rules Group 1

1. sepallength='(-inf-5.5]' petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' 47 ==> cluster=cluster3 47 conf:(1)

2. sepallength='(5.5-6.7]' petallength='(2.966667-4.933333]' ==> cluster=cluster1 34 conf:(1)

3. sepallength='(5.5-6.7]' petallength='(4.933333-inf)' 29 ==> cluster=cluster2 24 conf:(0.83)

explination

From these three results, we can see that the three clusters separate in the feature 'petallength'. Although it does not mean that ' petallength(-inf,-2.97) is indicating cluster3', but it does help us know each cluster.

Situation2:

Algorithm: SimpleKmeans with 3 clusters; Number of bins:5

Rules Group 1

1. sepallength='(-inf-5.02]' petalwidth='(-inf-0.58]' 27 ==> cluster=cluster3 27 conf:(1)

2. sepalwidth='(2.48-2.96]' petallength='(3.36-4.54]' petalwidth='(1.06-1.54]' 15 ==> cluster=cluster2 15 conf:(1)

3. sepalwidth='(2.96-3.44]' petalwidth='(2.02-inf)' 17 ==> cluster=cluster1 17 conf:(1)

4. sepallength='(5.02-5.74]' 18==> cluster=cluster2 18 conf:(1)

5. sepallength='(5.74-6.46]' petallength='(4.54-5.72]' ==> cluster=cluster1 20 conf:(1)

explanation

In this situation we have more bins for each feature. From rules 1,2 and 3, by comparing the results in the situation 1, we can find that the interval become more accurate which can help us know better. Meanwhile, in the situation 1, for the feature sepallength, cluster 1 and cluster 2 share the same region whereas in this situation they separate into two regions.

In this way, we would say this led to a better result. However, we also considered if 'more bins better result?'. The answer should be false. If we make each bin contains only one interval, then they will fail to reach the required support which will stop us.

Situation 3:

Algorithm: HierarchicalCluster with 3 clusters; linkType: Average; Number of bins: 3

Rules Group

1. petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' 50 ==> cluster=cluster1 50 conf:(1)

2. petallength='(2.966667-4.933333]' petalwidth='(0.9-1.7]' 48 ==> cluster=cluster2 48 conf:(1)

3. petallength='(4.933333-inf)' petalwidth='(1.7-inf)' 40 ==> cluster=cluster3 40 conf:(1)

Explanation:

In this situation, we set the number of bins to 3, the number of incorrectly clustered instances is 9, which is 6%. In this case, there are high-confidence association rules between different clusters and features, and the two important features are petallength and petalwidth.

Situation 4:

Algorithm: HierarchicalCluster with 3 clusters; linktype: Average; Number of bins: 5

Rules Group

1. petallength='(-inf-2.18]' petalwidth='(-inf-0.58]' 49 ==> cluster=cluster1 49 conf:(1)

2. sepallength='(5.74-6.46]' petallength='(4.54-5.72]' 27 ==> cluster=cluster2 27 conf:(1)

3. petallength='(5.72-inf)' 16 ==> cluster=cluster3 16 conf:(1)

Explanation:

In this case, the petallength has an effect on the association rules, and the association rules of cluster1 are similar to the previous example. When we increase the number of bins to 5, which produces a smaller interval, the sepallength feature replaces the role of petalwidth in situation 3. And the number of incorrectly clustered instances is 37, which is 24.6667 %. Increasing the number of bins did not give a better result.