TDDD41-732A75 Data Mining - Clustering and Association Analysis and Advanced Data Mining

Pranav Pankaj Chandode (prach896)

Varun Gurupurandar (vargu125)

Lab 2 Association Analysis - 1

Dataset

The lab instructions were followed to discretize the continuous attributes in the data using the respective filter.

Clustering

The application of the SimpleMeans clusterer was done with the parameters from the given instructions. The below figure shows crosstabulate of clusters and actual labels.

```
Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0  0 50 | Iris-setosa
 48  2  0 | Iris-versicolor
  7 43  0 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-virginica
Cluster 2 <-- Iris-setosa

Incorrectly clustered instances :       9.0       6      %
```

The results indicate that all Iris-setosa samples were correctly assigned to cluster 2. However, two Iris-versicolor samples were misclassified as Iris-virginica, while seven Iris-virginica samples were placed in the Iris-versicolor cluster. In total, nine instances were misclassified, accounting for 6% of the dataset.

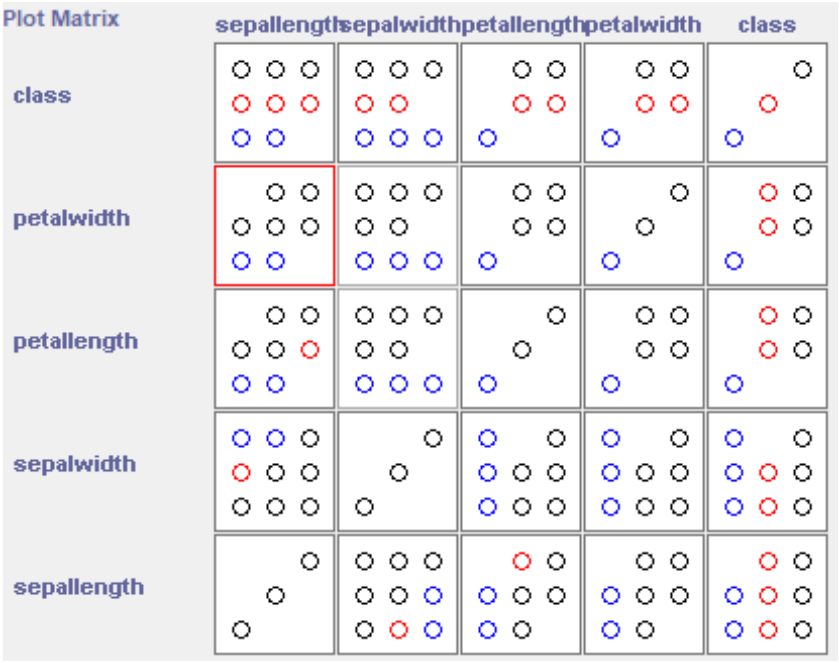
Association Analysis

The association analysis conducted still includes the class attribute, leading to at least 10 rules with a support of 50 and a confidence of 1. A large itemset of size 4 was identified, covering 47 instances of the Iris-setosa class. Additionally, the top 10 rules discovered are variations of attributes specific to the setosa class, where the class and its attributes function as both determinant and consequent. The below figure displays the large itemsets with size 2 or bigger and the 10 best rules found:

```
Size of set of large itemsets L(2): 10

Large Itemsets L(2):
sepallength='(-inf-5.5]' petallength='(-inf-2.966667]' 47
sepallength='(-inf-5.5]' petalwidth='(-inf-0.9]' 47
sepallength='(-inf-5.5]' class=Iris-setosa 47
petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' 50
petallength='(-inf-2.966667]' class=Iris-setosa 50
petallength='(2.966667-4.933333]' petalwidth='(0.9-1.7]' 48
petallength='(2.966667-4.933333]' class=Iris-versicolor 48
petalwidth='(-inf-0.9]' class=Iris-setosa 50
petalwidth='(0.9-1.7]' class=Iris-versicolor 49
petalwidth='(1.7-inf)' class=Iris-virginica 45

Size of set of large itemsets L(3): 5

Large Itemsets L(3):
sepallength='(-inf-5.5]' petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' 47
sepallength='(-inf-5.5]' petallength='(-inf-2.966667]' class=Iris-setosa 47
sepallength='(-inf-5.5]' petalwidth='(-inf-0.9]' class=Iris-setosa 47
petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' class=Iris-setosa 50
petallength='(2.966667-4.933333]' petalwidth='(0.9-1.7]' class=Iris-versicolor 47

Size of set of large itemsets L(4): 1

Large Itemsets L(4):
sepallength='(-inf-5.5]' petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' class=Iris-setosa 47

Best rules found:

 1. petalwidth='(-inf-0.9]' 50 ==> petallength='(-inf-2.966667]' 50    conf:(1)
 2. petallength='(-inf-2.966667]' 50 ==> petalwidth='(-inf-0.9]' 50    conf:(1)
 3. class=Iris-setosa 50 ==> petallength='(-inf-2.966667]' 50    conf:(1)
 4. petallength='(-inf-2.966667]' 50 ==> class=Iris-setosa 50    conf:(1)
 5. class=Iris-setosa 50 ==> petalwidth='(-inf-0.9]' 50    conf:(1)
 6. petalwidth='(-inf-0.9]' 50 ==> class=Iris-setosa 50    conf:(1)
 7. petalwidth='(-inf-0.9]' class=Iris-setosa 50 ==> petallength='(-inf-2.966667]' 50    conf:(1)
 8. petallength='(-inf-2.966667]' class=Iris-setosa 50 ==> petalwidth='(-inf-0.9]' 50    conf:(1)
 9. petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' 50 ==> class=Iris-setosa 50    conf:(1)
10. class=Iris-setosa 50 ==> petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' 50    conf:(1)
```

Visualization

The results of the analysis in the task above is visualized in the following matrix:



Describing clustering through association analysis

For all subsequent runs, we set the "car" parameter to True, ensuring the class is used as the consequent. Additionally, we enabled "outputItemsets" to True to obtain a more detailed output.

Variation 1: Kmeans 3 bins, 3 clusters

In the first run, we applied the parameters recommended in the instructions, using three bins in the discretize filter, three classes, and K-means for clustering.

This produced 43 rules that met the minimum support threshold of 0.2 and a minimum confidence of 0.9.

We show 2 rules for each cluster:

Cluster 1 (Versicolor):

sepallength='(5.5-6.7]' petallength='(2.966667-4.933333]' petalwidth='(0.9-1.7]' 33 ==> cluster=cluster1 33 conf:(1)

sepallength='(5.5-6.7]' petalwidth='(0.9-1.7]' 38 ==> cluster=cluster1 37 conf:(0.97)

A combination of sepal length between 5.5 and 6.7 and petal width between 0.9 and 1.7 suggests that the object belongs to the versicolor cluster. A less restrictive rule with only these two determinants has higher support but also includes instances from another class, resulting in a confidence of 0.97. By adding petal length between 3 and 5 to the rule, the support decreases slightly, but the confidence increases to 1.

Cluster 2 (Virginica):

petallength='(4.933333-inf)' petalwidth='(1.7-inf)' 40 ==> cluster=cluster2 40 conf:(1)

sepallength='(5.5-6.7]' sepalwidth='(2.8-3.6]' petalwidth='(1.7-inf)' 18 ==> cluster=cluster2 18 conf:(1)

A petal width greater than 1.7, combined with a petal length greater than 5, or a sepal length between 5.5 and 6.7 along with a sepal width between 2.8 and 3.6, indicates membership in cluster 2. Both rules have high confidence, though the rule with three determinants has lower support.

Cluster 3 (Setosa):

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

sepallength='(-inf-5.5]' sepalwidth='(2.8-3.6]' petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' 36 ==> cluster=cluster3 36 conf:(1)

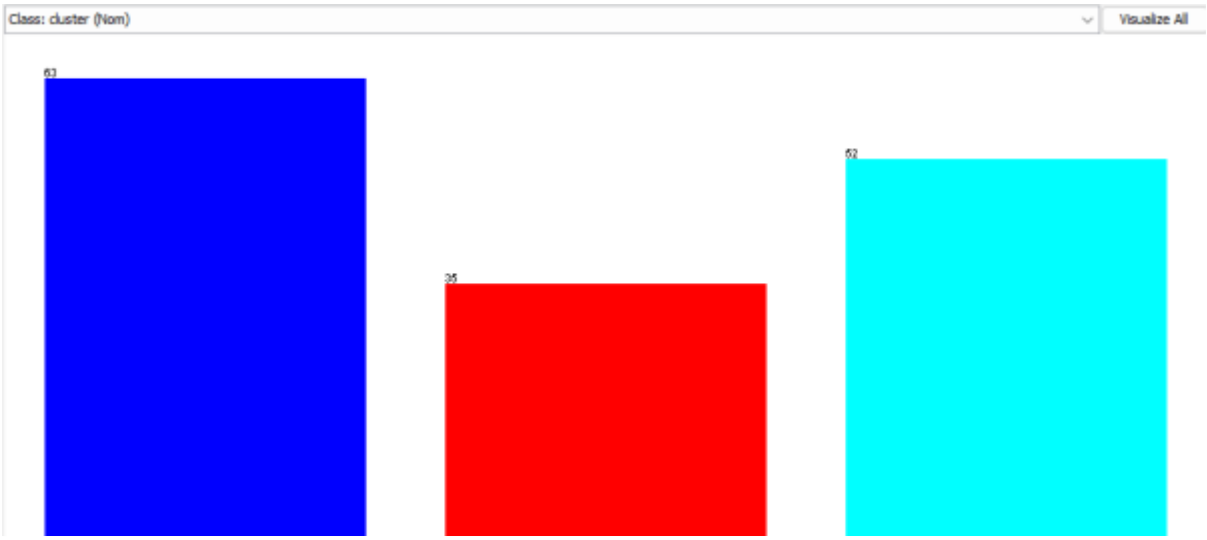
Both rules share the conditions of sepal length being less than 5.5 and petal length being less than 3, combining them with petal width less than 0.9 and sepal width between 2.8 and 3.6, respectively. This shared determinant strongly indicates the Setosa class.

The association analysis identifies rules with high support and confidence, enabling an accurate description of the clusters.

Variation 2: Kmeans 5 bins, 3 clusters

Since the three clusters align with the three actual classes, we retain this parameter for this run while continuing to use the K-means algorithm. The only change is increasing the number of bins to five.

This adjustment leads to the following clustering results:



The clustering does not perfectly align with the actual classes, as cluster 1 is larger than expected and cluster 2 is smaller.

We show 2 rules for each cluster one best and 1 worst with respect to minimum support and confidence.

Cluster 1:

sepallength='(5.74-6.46]' petallength='(4.54-5.72]' 27 ==> cluster=cluster1 27 conf:(1)

petallength='(4.54-5.72]' 47 ==> cluster=cluster1 44 conf:(0.94)

A petal length between 4.54 and 5.72 serves as an indicator for cluster 1. However, when used as the sole determinant, the confidence is not perfect. Adding sepal length between 5.74 and 6.46 improves the confidence to 1 but reduces the support to 27.

Cluster 2:

sepalwidth='(2.48-2.96]' petallength='(3.36-4.54]' 18 ==> cluster=cluster2 18 conf:(1)

petallength='(3.36-4.54]' petalwidth='(1.06-1.54]' 27 ==> cluster=cluster2 25 conf:(0.93)

A petal length between 3.36 and 4.54 frequently appears in the rules for cluster 2. When combined with a sepal width between 2.48 and 2.96, it achieves high confidence but has a support of only 18. Replacing sepal width with petal width between 1.06 and 1.54 increases the support to 25 but lowers the confidence to 0.93.

Cluster 3:

petallength='(-inf-2.18]' 50 ==> cluster=cluster3 50 conf:(1)

sepallength='(-inf-5.02]' 32 ==> cluster=cluster3 30 conf:(0.94)

A petal length smaller than 2.18 strongly defines the cluster, achieving high support and confidence. Sepal length smaller than 5.02 can also serve as an indicator for this cluster, but it has lower support and confidence.

Variation 3: EM Clustering 3 Clusters

We implemented EM clustering with 3 clusters and 3 bins for discretization, excluding the class attribute. We then applied the Apriori algorithm with a minimum support of 0.3, minimum confidence of 0.9, and set numRules to 10 for association analysis. The following association rules describe each cluster:

Cluster 0: No specific rule directly describing cluster 0 was identified with high confidence for cluster-specific attributes (antecedent excluding class), as the rules were primarily linked to cluster 2.

Cluster 1: No specific rule directly describing cluster 1 was identified with high confidence for cluster-specific attributes (antecedent excluding class), as the rules were primarily linked to cluster 2.

Cluster 2: petallength='(-inf-2.966667]' ==> cluster=cluster2 (confidence: 1, lift: 3), indicating this cluster includes flowers with small petals, likely Iris-setosa.

The rules for cluster 2 are highly reliable, with 100% confidence. However, no specific rules with high confidence were identified for clusters 0 and 1 without considering the class attribute. This is probably due to the probabilistic nature of EM clustering, which results in overlapping or less clearly defined cluster boundaries. In contrast to SimpleKMeans, EM generated fewer rules specific to each cluster and showed lower overall precision for clusters 0 and 1, as its probabilistic approach tends to merge cluster distinctions, making the association rules less clear.

Conclusion

SimpleKMeans clustering with 3 clusters and 3 bins for discretization generated the most accurate and meaningful association rules, achieving confidence values of up to 100% (e.g., *petallength='(-inf-2.966667]' petalwidth='(-inf-0.9]' ==> cluster=cluster0* conf:(1)). These rules effectively captured the distinct physical traits of the Iris species (small petals for Iris-setosa, large petals for Iris-virginica, and medium-sized petals for Iris-versicolor). On the other hand, EM clustering with 3 clusters produced less precise rules (confidence ranging from 82% to 88%), likely due to its probabilistic nature, which led to overlapping or less distinct cluster boundaries and decreased the clarity of the association rules. Similarly Kmeans with 5 bins was unable to produce good clustering and rules.