Association_lab_01

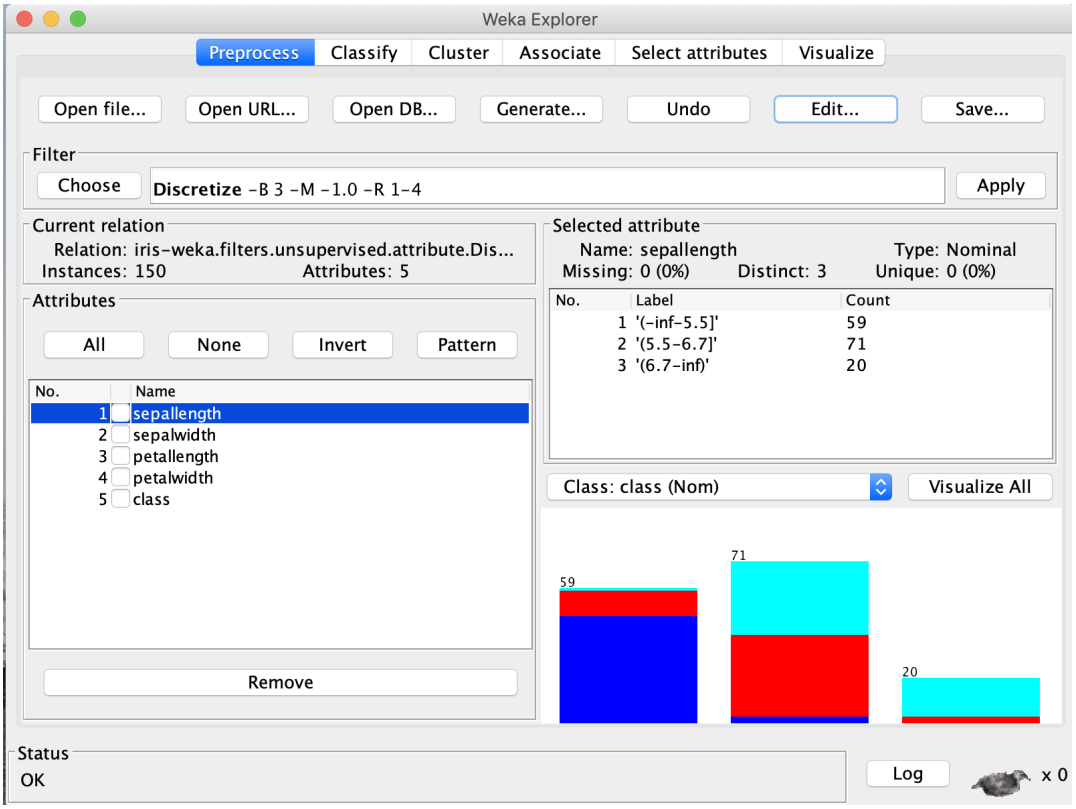
Thijs Quast (thiqu264), Lennart Schilling (lensc874)

28-2-2019

Contents

Dataset

After following the instructions to import and discretize the data, the following results are obtained:



As can be seen, the data is now discretized into three bins, additionally, the 5th attribute is not discretized. Resulting from this data processing, we are now able to perform association analysis. As an example, the three discrete values which are shown, are based on the sepallength attribute.

Clustering

We clustr the data based on the SimpleKmeans algorithm, with 3 clusters and a seed value of 10. Additionally, we ignore the class attribute.

```
=== Model and evaluation on training set ===


kMeans
======

Number of iterations: 3
Within cluster sum of squared errors: 96.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                                 Cluster#
Attribute                           Full Data           0                  1                  2
                                        (150)        (55)               (45)               (50)
=============================================================================================================
sepallength                        '(5.5-6.7]'   '(5.5-6.7]'        '(5.5-6.7]'        '(-inf-5.5]'
sepalwidth                         '(2.8-3.6]'   '(-inf-2.8]'       '(2.8-3.6]'        '(2.8-3.6]'
petallength     '(2.966667-4.933333]' '(2.966667-4.933333]' '(4.933333-inf)' '(-inf-2.966667]'
petalwidth                         '(0.9-1.7]'   '(0.9-1.7]'        '(1.7-inf)'        '(-inf-0.9]'



Time taken to build model (full training data) : 0.01 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       55 ( 37%)
1       45 ( 30%)
2       50 ( 33%)


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      %
```

Our data is divided into 3 clusters, we observe that the clusters 0, 1 and 2, are almost equal in size, namely, 37%, 30% and 33%, respectively.

Association analysis

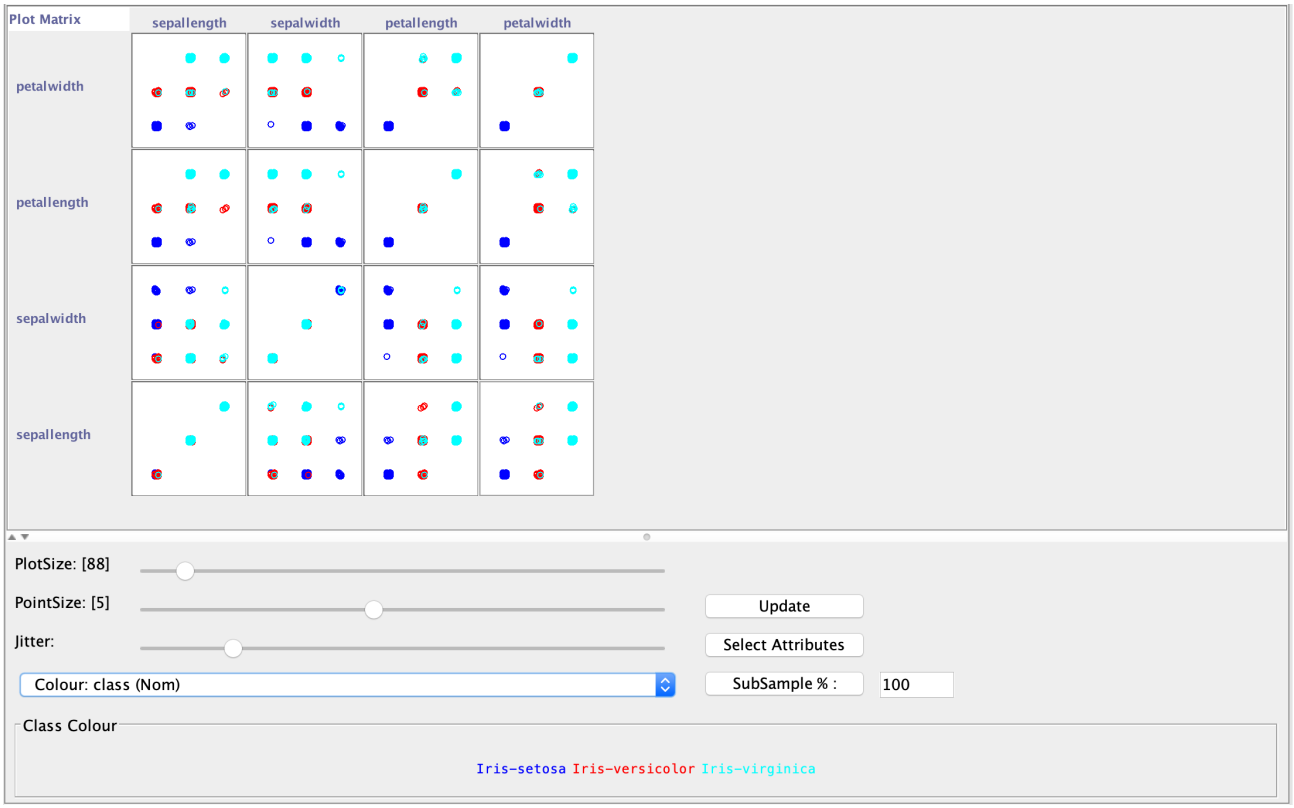
```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     iris-weka.filters.unsupervised.attribute.Discretize-B3-M-1.0-R1-4
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
              class
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.3 (45 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 14

Generated sets of large itemsets:

Size of set of large itemsets L(1): 13

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

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

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

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)
```

After performing the association analysis, based on the Apriori algorithm, with default properties, we obtained the above results. As can be seen, the algorithm uses a minimum support of 0.3. Additionally, 14 cycles were performed. Results are adequate. A total of $13 + 10 + 5 + 1 = 29$ frequent itemsets were identified.

Using these frequent itemsets, the best 10 rules which all satisfy to the minimum confidence criteria, this can be seen by the confidence behind the rules. For all rules the confidence is 1, which is greater than the minimum confidence of 0.9, setup as a criteria for the Apriori algorithm.

Also, we find that each association rule is supported by 50 observations, which is highlighted in the screenshot above.

Visualization



Above is a visualization of the data. As an example we can see that sepallength alone is not of great influence on the clusers as the crosstabulation of sepallength and sepallength on both axis show great overlap of the clusters. On the contrary, when we explore the crosstabulation of e.g. petallwidth is a better determinant of clustering, as this clustering based on this variable shows little overlap of the clusters.

Describing clustering through association analysis

3 bins, 3 clusters, SimpleKMeans

After creating an extra attribute for clusters, we set the numRules to 100, in order to create a sufficient amount of rules, therefore we hope to obtain rules that are accurate and in such a form that the antecedent does not contain the class attribute and the consequent only contains the cluster attribute. We have to find rules like this for all 3 clusters. After running the algorithm, we extract the following most important rules, which we believe cover all 3 clusters and have sufficient confidence:

Table 1: Rules for clusters

Rules	Cluster	Occurances	Confidence
petallength=(-inf-2.966667]	3	50	1
petalwidth=(-inf-0.9]	3	50	1
petallength=‘(2.966667-4.933333]’ petalwidth=(0.9-1.7]	1	48	1
sepalwidth=‘(-inf-2.8] petalwidth=’(0.9-1.7]	1	31	1
petallength=‘(4.933333-inf) petalwidth=’(1.7-inf)	2	40	1
sepalwidth=‘(2.8-3.6]’ petalwidth=’(1.7-inf)	2	29	1

As can be seen from the output above, several rules now determine to which cluster an observation belongs.

3 bins, 2 clusters, SimpleKMeans

Now, instead of dividing the data into 3 clusters, we choose to divide the data into 2 clusters.

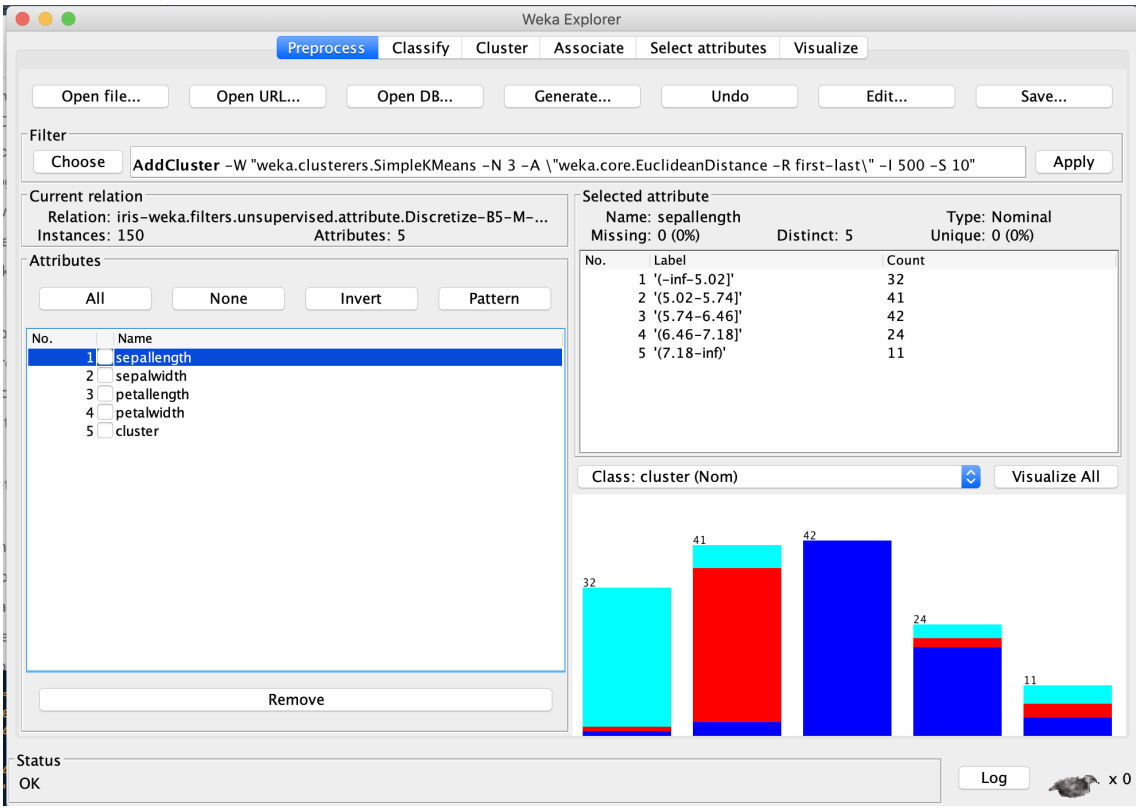
As can be seen from the output above, most observations fall in cluster 1. We now expect that most rules will only apply to cluster 1, simply because we expect there will not be much support for itemsets including observations for cluster 2. Because, this cluster has very few observations (17/150).

Our expectations are confirmed by the output of the association analysis. Actually, the best 25 rules found only apply to cluster 1.

Table 2: Rules for clusters

Rules	Clusters	Occurances	Confidence
sepallength=(5.5-6.7]	1	71	1
sepallength=(-inf-5.5]	1	59	1
sepallength=(6.7-inf) petallength=(4.933333-inf)	2	17	1

5 bins, 3 clusters, SimpleKMeans



As can be seen from the title of this paragraph, we now discretized the data into 5 bins. This actually means that the data is divided more specifically, therefore we expectt to find more important rules for each cluster. The rest of the parameters remain the default values of 3 clusters and SimpleKMeans clustering algorithm. The most important rules are shown below:

Table 3: Rules for clusters

Rules	Clusters	Occurances	Confidence
petallength=(4.54-5.72]	1	47	1
sepallength=(5.74-6.46]	1	42	1
sepallength=(-inf-5.02] petallength=(-inf-2.18]	3	28	1
sepallength=‘(-inf-5.02] petalwidth=’(-inf-0.58]	3	27	1
sepallength=(5.02-5.74] petallength=(3.36-4.54]	2	17	1
sepallength=(5.02-5.74] petallength=(3.36-4.54] petalwidth=(1.06-1.54]	2	15	1

Because we discretize the data into 5 bins instead of 3, the data is divided more specifically. Resulting from this, we get rules which are more specific to the attributes, when comparing for instance when the data was discretized into 3 bins. E.g. petallength=(-inf-2.966667] (3 bins) and petallength=(4.54-5.72] (5 bins).

Conclusion

From this lab, we conclude that both the parameters for the clustering and association have an important effect on the rules for the clusters found. Also, when we divide the data into 2 clusters (whilst we know that

the dataset actually has 3 clusters), we again find an important difference in the association rules found by Weka. When we discritize the data into 5 bins, we get rules which are more 'specific' in how they define the effect of the attributes. From this we conclude that one should always be careful and understand both the algorithms as well as the dataset that one is trying to analyze.