

Report: Lab 2 - 1

We decided to make variations to:

- Number of clusters
- Number of bins

For each experiment we followed the procedure in the lab material. In the analysis, we decided to use set numRules to 10 for each variation. The other setting for the Apriori algorithm was set to default values.

Baseline: 3 clusters, 3 bins

The best rules:

```
6. petallength='(-inf-2.966667]' 50 ==> cluster=cluster3 50 conf:(1)
10. petalwidth='(-inf-0.9]' 50 ==> cluster=cluster3 50 conf:(1)
```

Variations: number of clusters and number of bins

For each new attribute make from clustering, we used seed = 10.

When trying 5 clusters, none of the top 40 rules fulfilled the rule requirements, so to say none of the consequents were cluster attribute. This is since the clusters created was smaller than the original classes, so the support for rules including these clusters are not as high as rules containing class element or one of the three bins.

After this, we tried 5 clusters and 5 bins which generated rules with less support. Still, no rules would suffice due to reasons explained above. When having a larger number of bins, the intervals within the flower attributes becomes smaller. This makes the rules more precise compares to other rules with the same support and confidence but with larger intervals.

Now, 2 cluster and 5 bins were used. Here, many rules had high support and confidence but since the clusters are very general we do not get much information out of the rules. Since the intervals are smaller with 5bins than 3bins each bin is more likely to be associated with a cluster with high confidence.

Example:

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

Since 3 cluster seems to be the best choice, since we have 3 flowers. When using 8 bins, we get few rules with high support. Here are some interesting ones:

```
3. petallength='(-inf-1.7375]' 48 ==> cluster=cluster1 48 conf:(1)
5. petalwidth='(-inf-0.4]' 48 ==> cluster=cluster1 48 conf:(1)
13. petallength='(-inf-1.7375]' petalwidth='(-inf-0.4]' 46 ==> cluster=cluster1 46 conf:(1)
```

Since the high number of bins, the intervals are smaller. This means that the rules are more precise than other rules. awesome.