Classification: C4.5 Tree and Bayesian

The following program implements both C4.5 decision tree and Bayesian classification to classify against the mushroom data provided.

The provided datasets were created using the original mushroom dataset from UCI repository, with one attribute with missing values removed. The training dataset contains 7423 records and the test dataset 701 records. The first attribute is the class of each record and the rest 21 attributes are categorical attributes.

The result of the C4.5 implementation as well as the Weka ID3 result both gave a 100% accuracy:

```
Accuracy: 100.0%
[e, b, y, w, t, l, f, c, b, g, e, s, s, w, w, p, w, o, p, k, s, m] Class: e
[e, x, f, n, f, n, f, w, b, n, t, f, s, w, w, p, w, o, e, n, a, g] Class: e
[e, f, s, b, t, n, f, c, b, e, e, s, s, w, e, p, w, t, e, w, c, w] Class: e
[e, f, y, n, t, n, f, c, b, w, t, s, s, g, p, p, w, o, p, n, y, d] Class: e
[p, f, y, e, f, s, f, c, n, b, t, k, k, w, p, p, w, o, e, w, v, d] Class: p
```

Classifier output

```
Correctly Classified Instances         701               100      %
Incorrectly Classified Instances         0                 0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances              701

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         0          1         1         1          1        e
                 1         0          1         1         1          1        p
Weighted Avg.    1         0          1         1         1          1

=== Confusion Matrix ===

   a   b   <-- classified as
 357   0 |   a = e
   0 344 |   b = p
```

However this accuracy differed when compared to the Bayesian classification of the program to the Weka. The program output an accuracy of 99.86% compared to 95.72% from Weka:

```
Accuracy: 99.85734664764621%
[e, b, y, w, t, l, f, c, b, g, e, s, s, w, w, p, w, o, p, k, s, m] Class: e
[e, x, f, n, f, n, f, w, b, n, t, f, s, w, w, p, w, o, e, n, a, g] Class: e
```

Classifier output

```
Correctly Classified Instances         671               95.7204 %
Incorrectly Classified Instances        30                4.2796 %
Kappa statistic                          0.9143
Mean absolute error                      0.0428
Root mean squared error                  0.1751
Relative absolute error                  8.5652 %
Root relative squared error             35.0247 %
Total Number of Instances              701

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.997     0.084      0.925     0.997     0.96       0.998    e
                 0.916     0.003      0.997     0.916     0.955      0.998    p
Weighted Avg.    0.957     0.044      0.96      0.957     0.957      0.998

=== Confusion Matrix ===

   a   b   <-- classified as
 356   1 |   a = e
  29 315 |   b = p
```

When using the Random forest on 150 instances from the Iris dataset, the time taken to build model was 0.12 seconds with 100% accuracy against the training set. Regardless of the size of the data, the C4.5 classifier against 701 instances of the mushroom set took 0.08 seconds with 100% accuracy. This is significantly faster compared to Random forest. Preprocessing could help significantly by removing any extreme cases or any missing attributes and smoothing out the data. Furthermore, pruning could be effective for unseen data by increasing its accuracy, as well as mitigating overfitting.

Few of the lessons learned is that creating the data structure and the algorithm for decision tree is extremely difficult compared to the implementation of Bayesian classification. The calculating the entropy and deciding the splits through that manner as proven to be the most difficult in programming the classifier. However, it proved to me a much more accurate classifier compared to Bayesian. However what is useful is that to some degree, attributes that do not matter will not be chosen as the splitting attribute, and will eventually get pruned out. So it is much more tolerant to nonsense.