**B-659 – Applied Machine Learning– PA- 1**

**Team –Prashanth Balasubramani, Siddharth Jayasankar**

1. Start from depth = 1 and go to different depths (2,4,6,8...,16). For each depth, compute the error (the number of misclassifications) on the test set. Plot a learning curve with the depth of the tree on the x-axis and the accuracy on the y-axis.

[Ans]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Depth** | **Misclassifications**  **In Train 1 -Test 1** | **Accuracy in Train 1 -Test 1** | **Misclassifications**  **In Train 2 -Test 2** | **Accuracy in Train 2 - Test 2** | **Misclassifications**  **In Train 3 -Test 3** | **Accuracy in Train 3 - Test 3** |
| 1 | 108 | 0.75 | 142 | 0.671 | 84 | 0.805 |
| 2 | 108 | 0.75 | 162 | 0.625 | 12 | 0.972 |
| 3 | 72 | 0.833 | 146 | 0.662 | 24 | 0.944 |
| 4 | 72 | 0.833 | 152 | 0.662 | 20 | 0.953 |
| 5 | 72 | 0.833 | 131 | 0.696 | 44 | 0.891 |
| 6 | 48 | 0.888 | 94 | 0.782 | 44 | 0.891 |
| 7 | 40 | 0.907 | 66 | 0.847 | 44 | 0.891 |
| 8 | 40 | 0.907 | 58 | 0.866 | 45 | 0.885 |
| 10 | 32 | 0.925 | 58 | 0.866 | 44 | 0.891 |
| 12 | 32 | 0.925 | 58 | 0.866 | 44 | 0.891 |
| 14 | 32 | 0.925 | 58 | 0.866 | 44 | 0.891 |
| 16 | 32 | 0.925 | 58 | 0.866 | 44 | 0.891 |

1. Report the learned decision tree (depth 1 and depth 2) and report the confusion matrix for these two depths (A confusion matrix has the true label as rows and predicted labels in the columns. Each entry of the matrix is the number of examples. In a binary case, the top left corner is the number of negative examples correctly classified and the bottom right is the number of positives correctly classified).

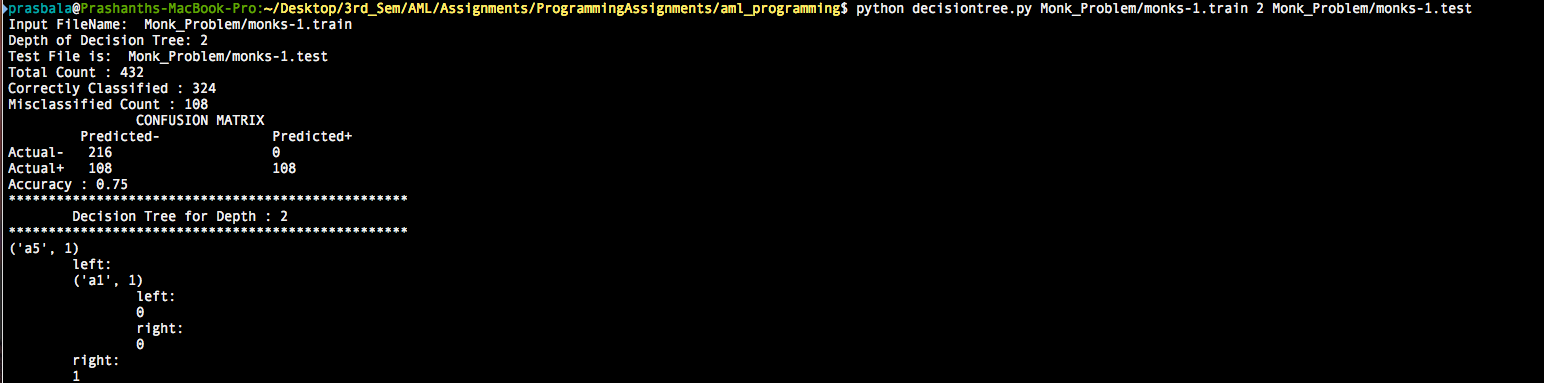
[Ans]

Below are the screen shots of the confusion matrix and the decision trees.

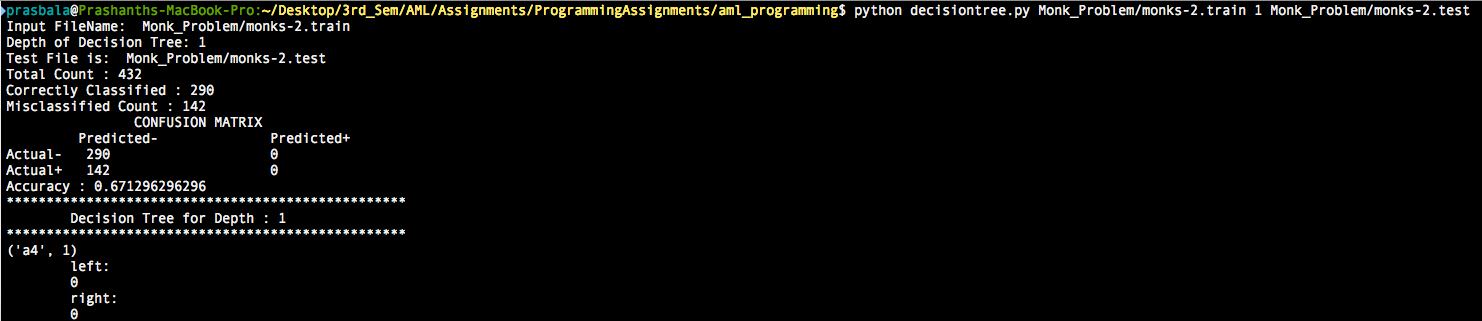
**Note:** In the decision trees displayed below, left branches are false branches and right branches are true branches



Confusion Matrix and depth 1 tree for classifier trained on Monks-1.train and tested on Monks-1.test



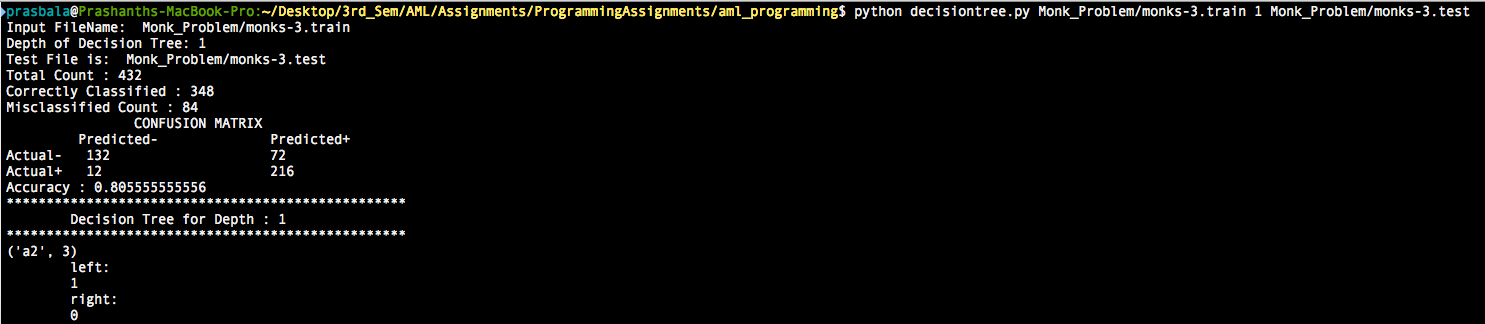
Confusion Matrix and depth 2 tree for classifier trained on Monks-1.train and tested on Monks-1.test



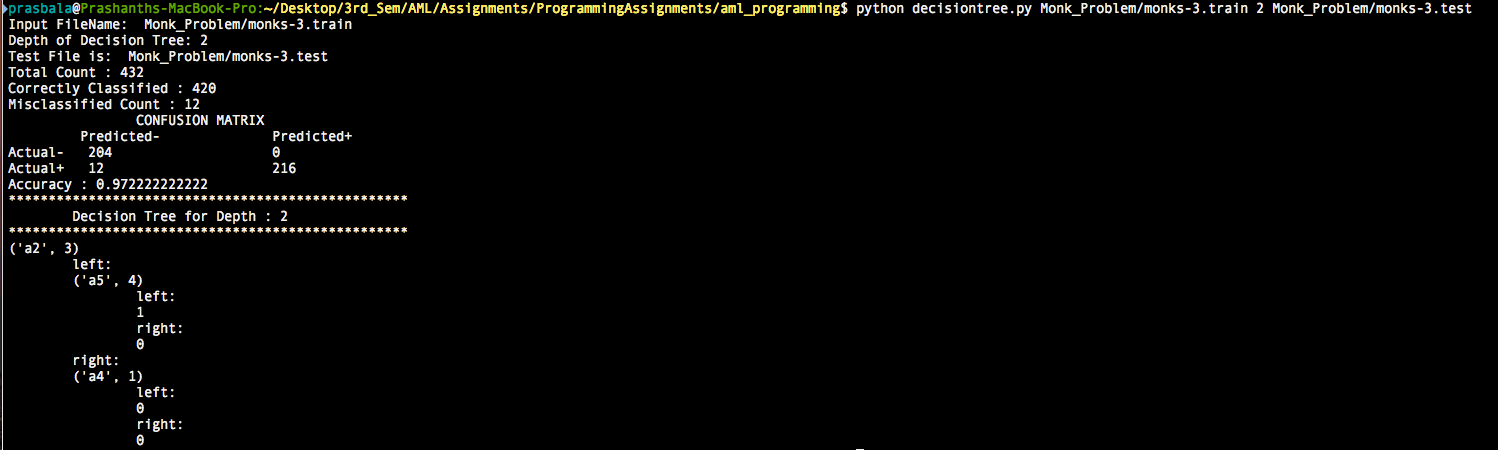
Confusion Matrix and depth 1 tree for classifier trained on Monks-2.train and tested on Monks-2.test



Confusion Matrix and depth 2 tree for classifier trained on Monks-2.train and tested on Monks-2.test



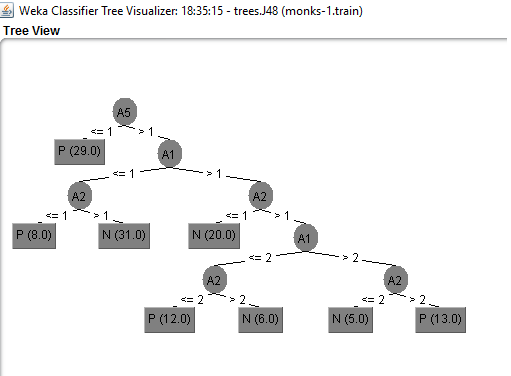
Confusion Matrix and depth 1 tree for classifier trained on Monks-3.train and tested on Monks-3.test



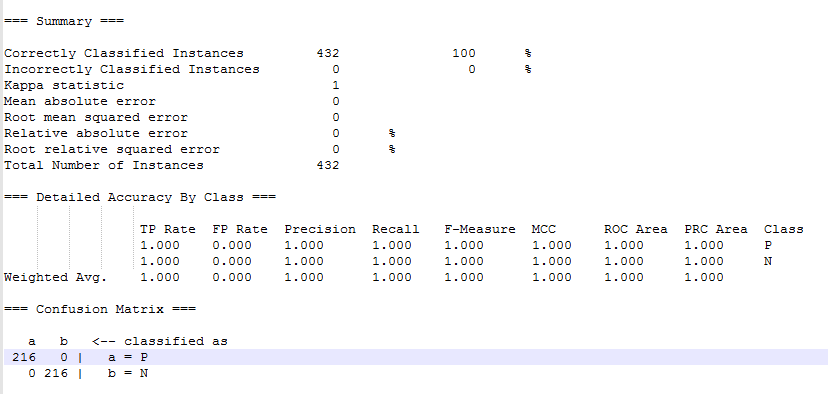
Confusion Matrix and depth 2 tree for classifier trained on Monks-3.train and tested on Monks-3.test

1. Now, use Weka's default decision tree (J48) algorithm on this training set to learn a decision tree. Report the tree and the confusion matrix on the test set. Do not change the default parameters of Weka.

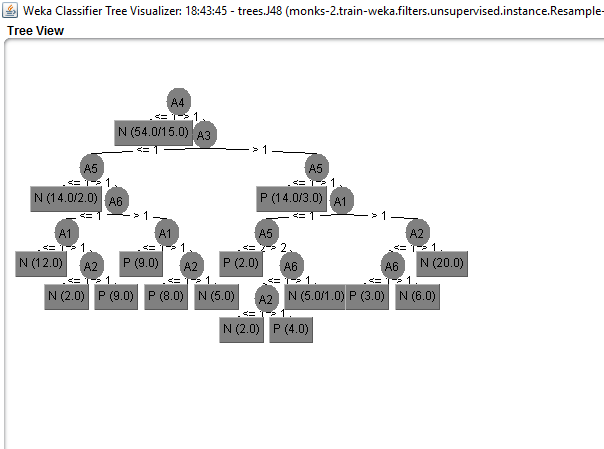
[Ans]



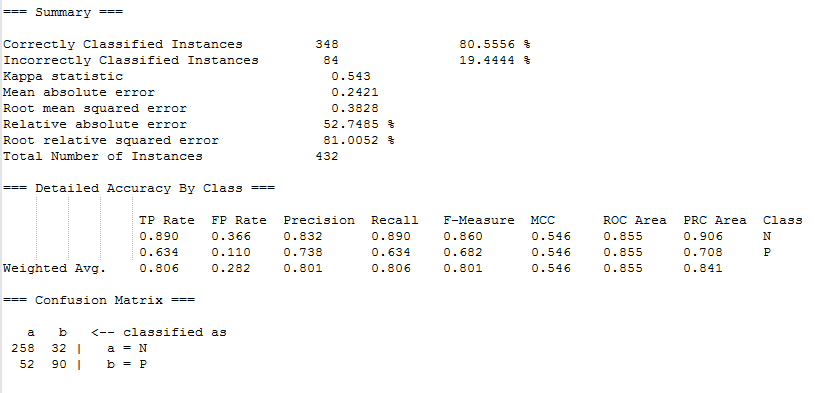
Decision tree generated on Monks-1.train by Weka



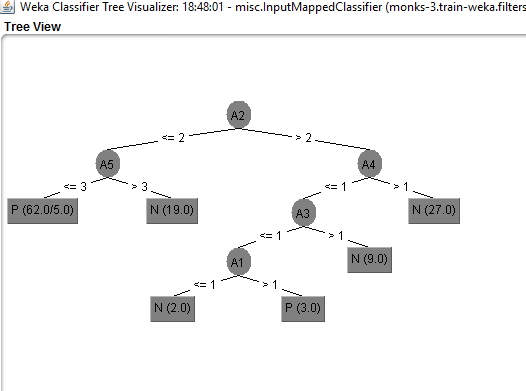
Summary and Confusion Matrix generated by Weka for Monks-1.train and tested on Monks-1.test



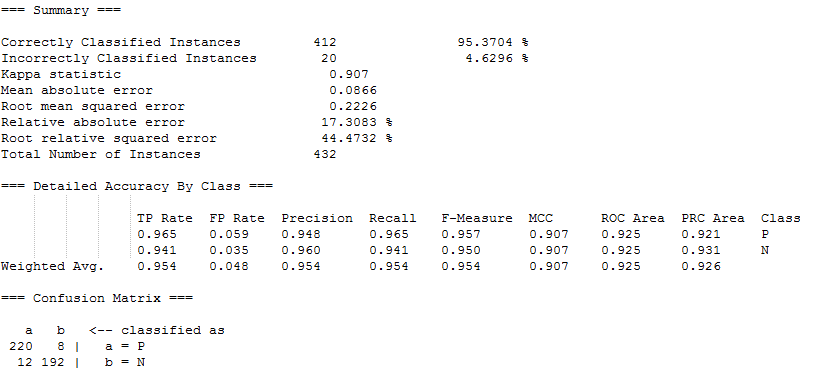
Decision tree generated on Monks-2.train by Weka



Summary and Confusion Matrix generated by Weka for Monks-2.train and tested on Monks-2.test



Decision tree generated on Monks-3.train by Weka

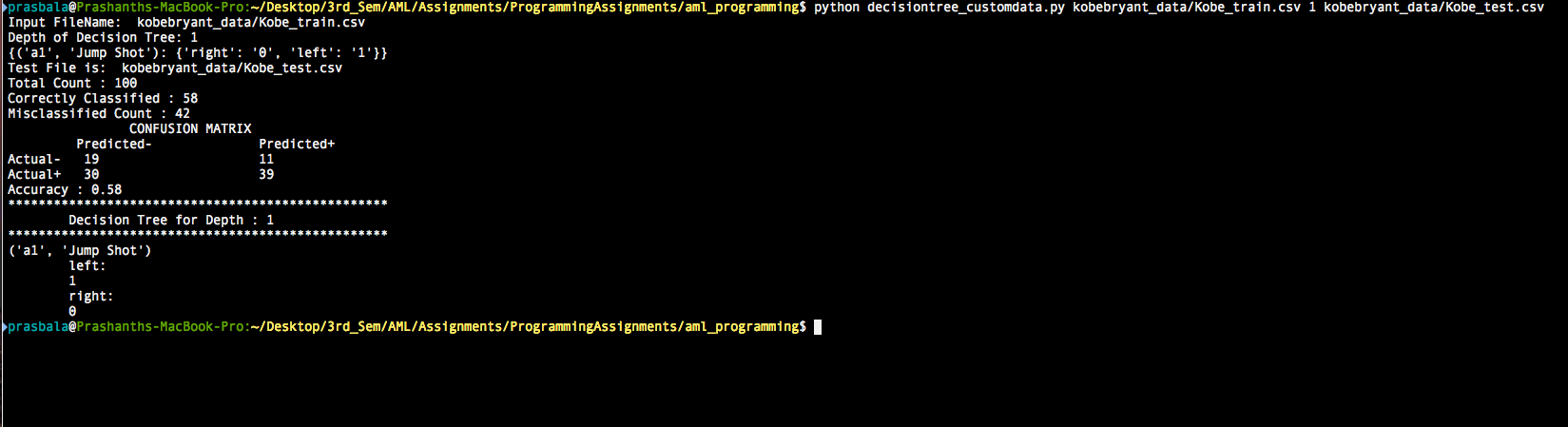


Summary and Confusion Matrix generated by Weka for Monks-3.train and tested on Monks-3.test

**Note:** The class values are changed from 0 -> N and 1 ->P in the above Weka screen shots. This was done to facilitate Weka recognizing the class variable as nominal values.

1. Repeat steps 2 and 3 with your own data set and report the confusion matrices.

[Ans]



Confusion Matrix and depth 1 tree for classifier trained on kobe Bryant train and test data set.

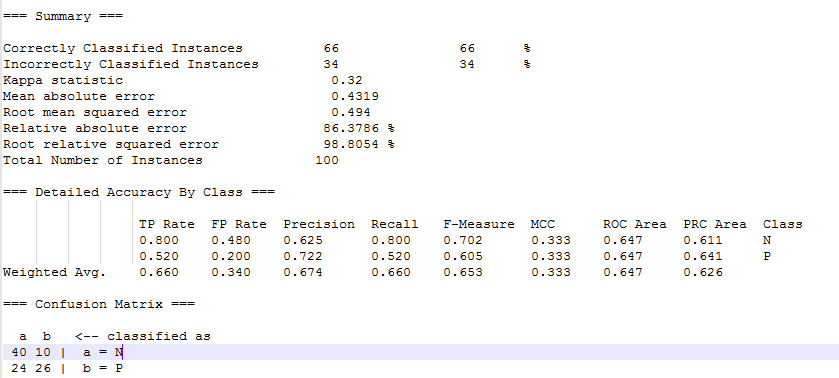


Confusion Matrix and depth 2 tree for classifier trained on kobe Bryant train and test data set.

The decision tree generated by Weka for our own data set is very large to be printed clearly. A text version of the tree is present in this text file.



Have include the above decision tree as a separate file also.



Summary and Confusion Matrix generated by Weka for kobe\_bryant.train and tested on kobe\_bryant.test