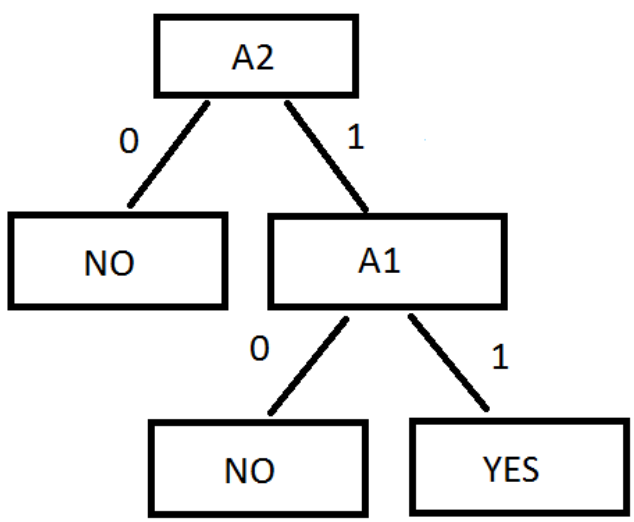
Antonio Rafael V. Umali
CS 534
HW 5 Writeup

I.

18.3 The algorithm will not return the exact correct tree, rather a tree that is logically equivalent to the correct tree. This means that the algorithm will return a tree that is defined on the same set of attributes as the correct tree and both trees will agree on all possible examples.

As shown in Chapter 18, the decision tree learning algorithm always looks at the provided examples or the test data. Importance was then defined in terms of information gain, or the inverse of entropy. Entropy was defined as the uncertainty of a random variable, and acquiring more information reduces entropy and thus increases information gain. Increasing the training set size to infinity will allow the algorithm to be able to assign importance better and thus create a tree which will be logically equivalent to the true decision tree (or function).

18.6



The following calculations were made:
$B(q) = -(q\log_2 q + (1-q)\log_2(1-q))$
Step 1:

$Gain(A1) = B(2/5) - (4/5 * B(2/4) + 1/5 * B(0/1)) = 0.1710$
$Gain(A2) = B(2/5) - (3/5 * B(2/3) + 2/5 * B(0/2)) = 0.4200$
$Gain(A3) = B(2/5) - (2/5 * B(1/2) + 3/5 * B(1/3)) = 0.0200$

A2 chosen.

Step 2:

$$Gain(A2) = B(2/3) - (2/3 * B(2/2) + 1/3 * B(0/1)) = 0.9183$$
$$Gain(A3) = B(2/3) - (1/3 * B(1) + 2/3 * B(1/2)) = 0.2516$$

A1 chosen.

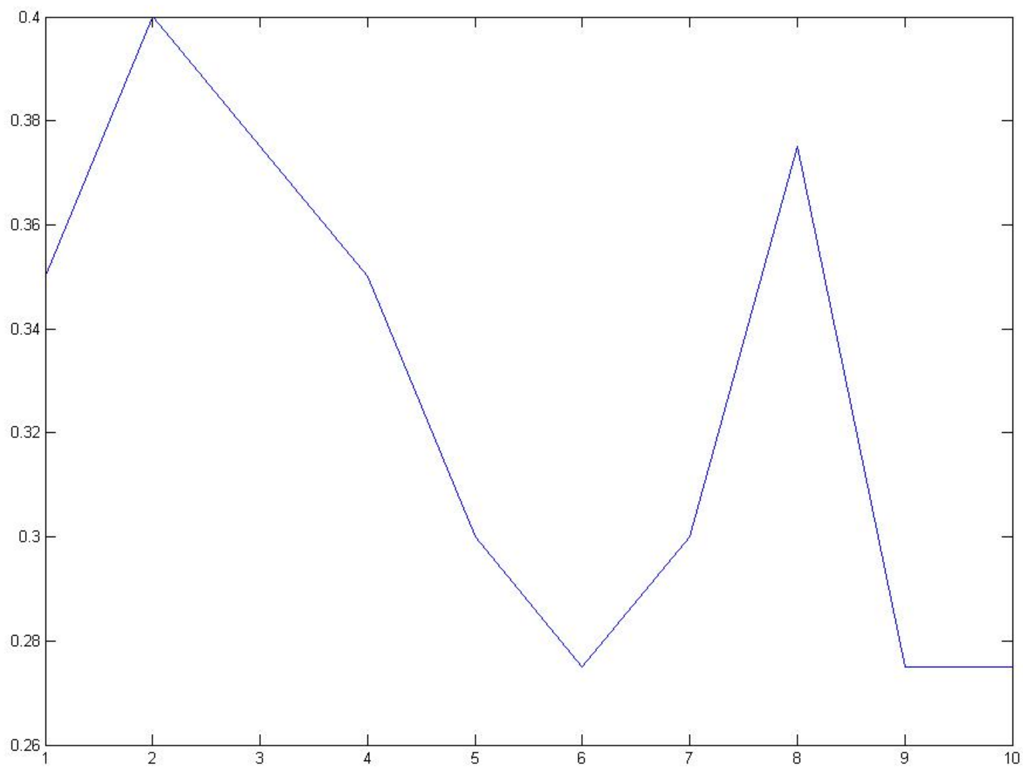
Step 3:

$$Gain(A3) = B(0) - (½*B(1) + ½*B(1)) = 0$$

No need to include A3

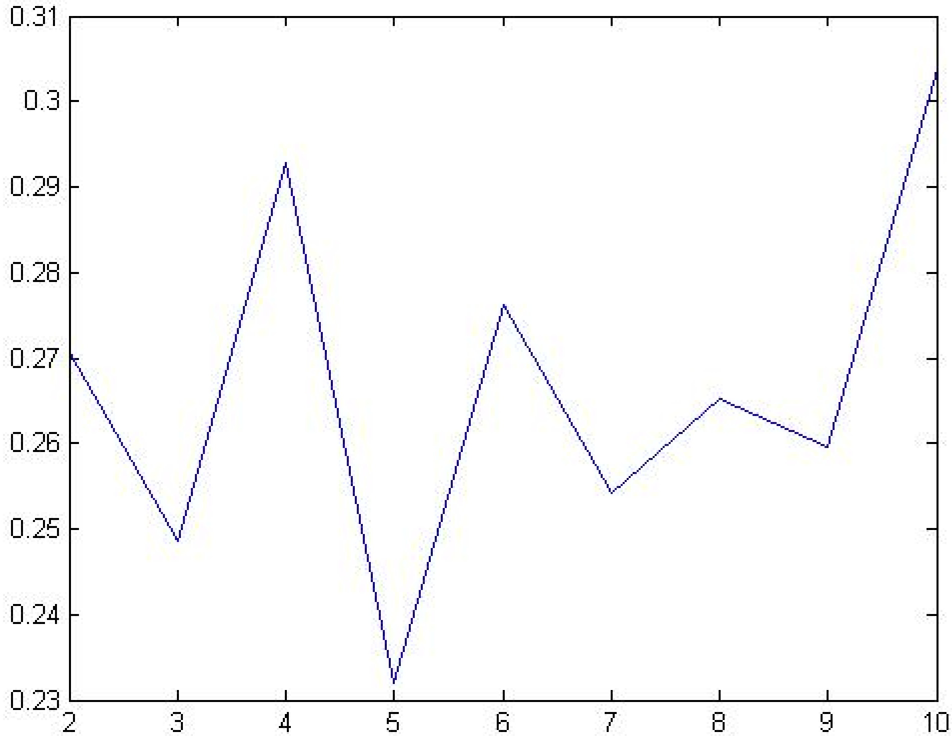
II.

1.



I believe that the KNN plot gradually gets better from K = 1 to 5 and then spikes back up at K = 6 then returns to normal at K = 9 is because of the nature of the data. The accuracy of the KNN depends entirely on how the data is clustered; if at K = 6 there are more 1's than 0's around most points then the algorithm will lean towards classifying more points as 1's. Thus the K-plot represents, in general, how the training data is clustered and how good these clusters are for classifying.

2.



I did this plot using 100,000 epochs for the ANN. There is an unusual spike in the error rate at N=6 and then up again at 10. Though it can be seen that the ANN errors are, compared to the KNN's errors, much more stable. I think that this is because classification of the data can be achieved best via the use of a function very close to a straight line, maybe with a bit of peaks depending on the degree. Adding more neurons (beyond 2) simply increases the degree of the function the ANN is generating. Extending beyond 2 neurons might not hurt the classifier a lot, since the curves and peaks in the function might cause it to arc over or under the proper boundary and not drastically over or underfit the data. However, I can see that further increasing the number of neurons may cause the ANN to overfit the data more and more.

3. In terms of accuracy, I would say that the ANN would be better for this data set if time is not a concern. The ANN showed a more uniform error rate plot compared to the KNN, though this was achieved using 100,000 epochs/iterations. The ANN performed poorly when using very few epochs and was outmatched by the KNN when they were

set to run at more or less the same speed (done via manual tuning of the epochs). As for which one would benefit with more data, I think that the KNN would benefit more. As mentioned earlier, the best function that the ANN can fit for the data is a straight line, and more data would only vary the slope and intercepts of said line by a little. More data for the KNN, however, would give it more “votes” and thus increase its ability to classify new points. However, finding the right k for more data is another matter.