CS181 - Homework 1

Lucas Freitas and Angela Li

February 16, 2013

1. (a) i. For A:

$H(label)$: $\#(yes) = 4$, $\#(no) = 3$, so:

$$p(yes) = \frac{4}{7}$$

$$p(no) = \frac{3}{7}$$

$$H(label) = \frac{4}{7}log_2(\frac{7}{4}) + \frac{3}{7}log_2(\frac{7}{3}) \approx 0.985$$

$H(label|A = true) = 1$, since it is equally likely to be positive or negative given that $A = true$.

$H(label|A = false)$: $\#(yes) = 2$, $\#(no) = 1$, so:

$$p(yes) = \frac{2}{3}$$

$$p(no) = \frac{1}{3}$$

$$H(label|A = false) = \frac{2}{3}log_2(\frac{3}{2}) + \frac{1}{3}log_2(3) \approx 0.918$$

Therefore:

$$H(label|A) = p(A = true)H(label|A = true) + p(A = false)H(label|A = false)$$

$$H(label|A) = \frac{4}{7}H(label|A = true) + \frac{3}{7}H(label|A = false) \approx 0.965$$

Finally:

$$I(label; A) = H(label) - H(label|A) \approx 0.02$$

ii. For B:

$H(label|B = true) = 1$, since it is equally likely to be positive or negative given that $B = true$.

$H(label|B = false)$: $\#(yes) = 3$, $\#(no) = 2$, so:

$$p(yes) = \frac{3}{5}$$

$$p(no) = \frac{2}{5}$$

$$H(label|B = false) = \frac{3}{5}log_2(\frac{5}{3}) + \frac{2}{5}log_2(\frac{5}{2}) \approx 0.971$$

Therefore:

$$H(label|B) = p(B = true)H(label|B = true)+p(B = false)H(label|B = false)$$

$$H(label|A) = \frac{2}{7}H(label|A = true) + \frac{5}{7}H(label|A = false) \approx 0.979$$

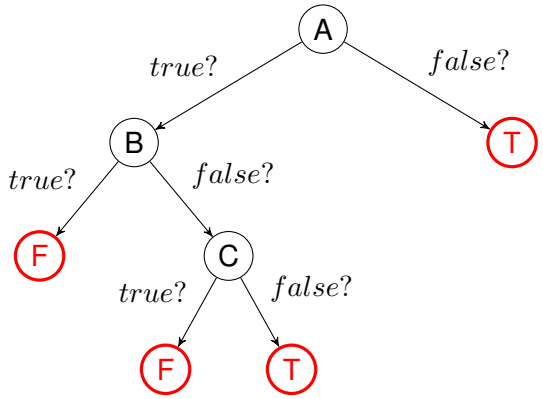
Finally:

$$I(label; B) = H(label) - H(label|B) \approx 0.006$$

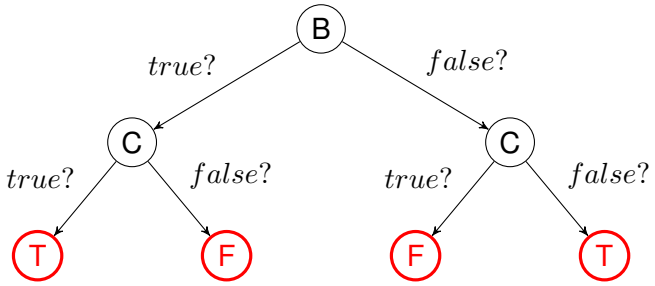
Thus, ID3 would split on A, since that gives a higher information gain. An argument in favor of splitting on A is that it is a more "extreme" split with more disparate probabilities, meaning it will more resolutely divide the data. An argument in favor of splitting on B is that it will classify a larger subset well.

This shows that the inductive bias of ID3 is to prefer more "extreme" splits which more resolutely divide the data.

(b) From looking at the data it is fairly obvious that in figuring out the first attribute to split on, B, C, and D have the same information gain. Since we calculated that A has a higher information gain than B, we'll first split on A. When A is 1, we can split on either B or C, which still have the same information gain at this step; we can recurse in a similar fashion from there, preferring binary splits. When A is 0, we can again split on either B or C, but (assuming we pick B) we run into the base case of no more information gain after traveling down the 0 path. So we can set that leaf to the most common label, 1.



(c) This tree has a training error of 1/7, just like the ID3 tree produced above. It shows us that ID3 certainly does not always generate the shortest trees, and specifically that preferring higher information gain for individual attributes at a time (and not being able to look ahead at combinations of attributes) often does not produce the most concise tree.



2. (a) The average cross-validated training and test performance over the ten trials for the non-noisy dataset is 0.87 and 0.78 for the noisy dataset.

(b) i. Graphs outputted from the Python script:

```
$ python main.py
```

ii. A. For non-noisy data: the performance of the train data initially oscillates, but then increases almost monotonically, reaching 100% performance when the validation set size reaches 78. The test data performance oscillates a lot, reaching a maximum at validation set size 21, with performance 89%, which is slightly higher ($\approx 2.3\%$ increase) than the one obtained without pruning. The performance then decreases, reaching 76% performance for a validation set of size 80.

B. For noisy data: the performance of the train data initially oscillates, but then also increases almost monotonically, reaching $\approx 98.3\%$ performance when the validation set size reaches 78. The test data performance oscillates a lot,

reaching a maximum at validation set size 8, with performance 82%, which is slightly higher ($\approx 5.1\%$ increase) than the one obtained without pruning. The performance then decreases, reaching 71% performance for a validation set of size 80.

iii. For some values of the validation set, yes. As mentioned in the previous item, for the non-noisy data we reach an 89% performance at validation set size 21, which is a $\approx 2.3\%$ increase in the performance of ID3, while for noisy data the maximum occurs at validation set size 8 with 82% performance, which is a $\approx 5.1\%$ increase) in the performance of ID3. If the validation set is too large, however, the ID3 performance decreases ($\approx 12.6\%$ performance decrease for non-noisy and 8.97% for noisy).

iv.

3.

4.