CS181 Assignment 1

Joy Ming and Alisa Nguyen (9 February 2013)

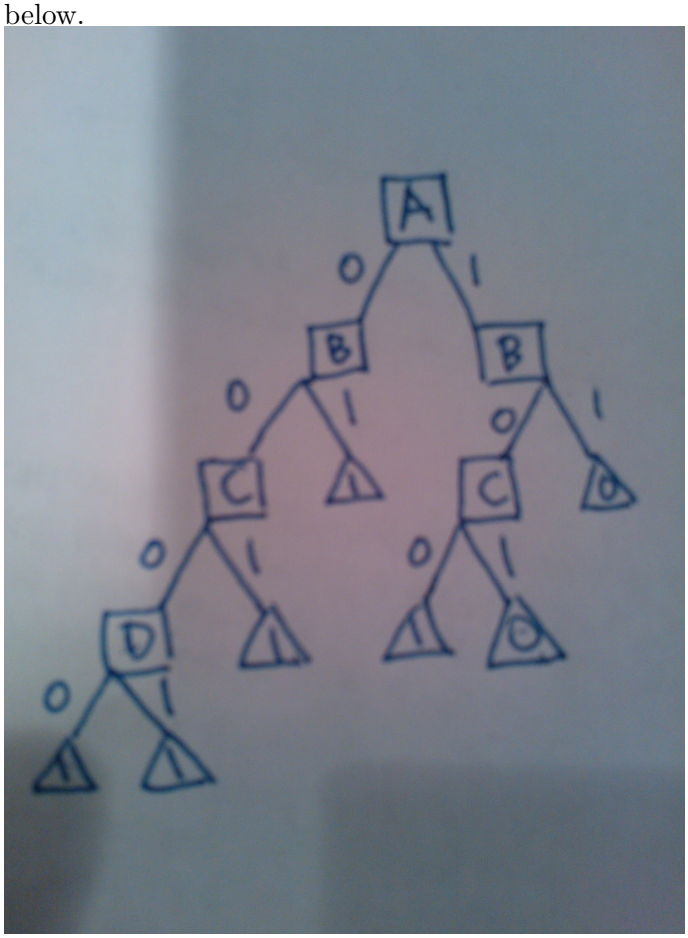
1. Decision Trees and ID3

   (a) ID3 will chose to split on $\boxed{\text{A}}$ because it has a higher information gain.

   - Splitting on A will have an information gain of $Gain(X_k, A) = H(A) - Remainder(X_k, A) = \boxed{0.025}$, where $H(A) = \frac{3}{7}\log_2\frac{7}{3} + \frac{4}{7}\log_2\frac{7}{4} = 0.985$, and $Remainder(X_k, A) = \frac{4}{7}(\frac{2}{4}\log_2 2 + \frac{2}{4}\log_2 2 + \frac{3}{7}(\frac{2}{3}\log_2\frac{3}{2} + \frac{1}{3}\log_2 3) = 0.96$
   - Splitting on B will have an information gain of $Gain(X_k, B) = H(B) - Remainder(X_k, B) = \boxed{0.005}$, where $H(B) = \frac{4}{7}\log_2\frac{7}{4} + \frac{2}{7}\log_2\frac{7}{2} = 0.985$, and $Remainder(X_k, B) = \frac{2}{7}(\frac{1}{2}\log_2 2 + \frac{1}{2}\log_2 2 + \frac{5}{7}(\frac{3}{5}\log_2\frac{5}{3} + \frac{2}{5}\log_2\frac{5}{2}) = 0.98$

   This example shows that ID3 has an inductive bias of strongly preferring extreme partitions and larger subsets. In this case, looking at the results of the of when both A and B are true, they have the same 1:1 ratio of positive and negative outputs, but A is preferred because it has two data points for each whereas B only has one. IS THIS TRUE??

   (b) In this example, a tree that could be formed would split first on A, then B, then C, as shown below.




   This tree splits first at A, without a tie.
   Then following A = 1, both B and C have the same information gain so we split on B to proceed alphabetically.

   - Where B = 1 there is only one output, Label = 0, so we return a leaf.

- Where B = 0 there are different outputs so we split on C, which returning two leaves: when C = 1, Label = 0 and when C = 0, Label = 1.

Following A = 0, both B and C have same information gain, and we split on B for the same reasons as above.

- When B = 1, there is only one value for Label, or Label = 1.
- When B = 0 we split on C because it has higher information gain than D. Since there are no cases where A = 0, B = 0 and C = 1, we return the majority Label = 1. Since there are still too many possible labels for C = 0, we split on D.
  - Since there are no examples where A = 0, B = 0, C = 0, D = 0, we return the majority Label = 1.
  - Since there are equal number of examples for both possible labels for D = 0, we broke this tie be returning the majority label, in this case, Label = 1.

(c) By eyeballing the data we can see first and foremost since there are equal numbers of B = 0 and C = 0 as well as B = 1 and C = 1 respectively for the A = 0 case, that splitting on the C branch is unnecessary. Therefore, creating a tree that, for the A = 0 path splits only on B and then D would create a tree with the same training error as the one produced by ID3. From this example we can learn that the ID3 algorithm has inductive bias, especially because it is greedy and only chooses the single attribute with the highest information gain.

2. ID3 with Pruning

(a) The average cross-validated training performance was:

- Non-noisy: Training $\boxed{1.0}$ and test $\boxed{0.87}$.
- Noisy: Training $\boxed{0.98}$ and test $\boxed{0.78}$.

(b) After the pruning function:

i. Graph

ii. The cross-validated performance of the validation set pruning improves at first, as the valdiation increases from 1, peaks at a point around size 40 to 60, and then worsens in performance as the validation set size becomes too large and overfitting becomes an issue.

iii. The validation set pruning improves the cross-validated performance of ID3 on these data for all the data points leading up to the peak when comparing against validation-size. After the peak, pruning gives us similar and sometimes slightly worse results than the cross-validated performance of ID3. On the nosy data, the average cross-validated test perfromance with pruning on the non-noisy dataset is 0.8599 and without pruning 0.855.

iv. Overfitting is an issue for these data, as evidenced by the dropoff after a peak when the validation set size gets too large. ELABORATE.

3. Boosting

(a) The weighted entropy of the set can be calculated:
$W = 0.5 + \frac{0.5}{N-1}(N-1) = 1$ $H = 0.5 \log_2 \frac{1}{0.5} + 0.5 \log_2 \frac{1}{0.5} = \boxed{1}$

i. Analyze the effectiveness of boosting:

A. Effect of maximum depth on cross validated boosting in noisy and not noisy data

| Noisy? | Max depth | $R = 10$ | $R = 30$ |
|---|---|---|---|
| N | 1 | 0.82 | 0.84 |
| NN | 1 | 0.89 | 0.91 |
| N | 2 | 0.81 | 0.79 |
| NN | 2 | 0.87 | 0.87 |

For each given set of data that is the same noisyness and number of rounds, it seems the greater the maximum depth the less accurate the created learner is. This is in part because the bigger the tree, the less of a "weak" learner it is. In this example, the greater the depth the more splits the tree will go through and thus will create a higher probabilitiy of overfitting. NEEDS MORE BUZZ WORDS

B. Effect of number of boosting rounds on cross-validated performance of decision trees
(graph)
This is expected based on our theoretical discussion of boosting in class because

C. Comparing cross-validated test performance of boosting with ID3 with and without pruning.

D. Comparing cross-validated training and test performance for boosting with weak learners of depth 1 over a number of rounds in [1, 15].
(graph)
The relationship between training and test performance shows that

4. Tree Analysis