CS 5350/6350: Machine Learining Spring 2019

Homework 1

Handed out: 25 January, 2019
Due date: 11:59pm, 10 Feb, 2019

- You are welcome to talk to other members of the class about the homework. I am more concerned that you understand the underlying concepts. However, you should write down your own solution. Please keep the class collaboration policy in mind.
- Feel free discuss the homework with the instructor or the TAs.
- Your written solutions should be brief and clear. You need to show your work, not just the final answer, but you do *not* need to write it in gory detail. Your assignment should be **no more than 15 pages**. Every extra page will cost a point.
- Handwritten solutions will not be accepted.
- *Your code should run on the CADE machines.* You should include a shell script, `run.sh`, that will execute your code in the CADE environment. Your code should produce similar output to what you include in your report.

  You are responsible for ensuring that the grader can execute the code using only the included script. If you are using an esoteric programming language, you should make sure that its runtime is available on CADE.
- Please do not hand in binary files! We will *not* grade binary submissions.
- The homework is due by **midnight of the due date**. Please submit the homework on Canvas.

1 Decision Tree [40 points + 10 bonus]

x_1	x_2	x_3	x_4	y
0	0	1	0	0
0	1	0	0	0
0	0	1	1	1
1	0	0	1	1
0	1	1	0.	0
1	1	0	0	0
0	1	0	1	0

Table 1: Training data for a Boolean classifier

1. [7 points] Decision tree construction.

(a) [5 points] Use the ID3 algorithm with information gain to learn a decision tree from the training dataset in Table 1. Please list every step in your tree construction, including the data subsets, the attributes, and how you calculate the information gain of each attribute and how you split the dataset according to the selected attribute. Please also give a full structure of the tree. You can manually draw the tree structure, convert the picture into a PDF/EPS/PNG/JPG format and include it in your homework submission; or instead, you can represent the tree with a conjunction of prediction rules as we discussed in the lecture.

$$H_y = -\frac{2}{7}log_2(\frac{2}{7}) - \frac{5}{7}log_2(\frac{5}{7}) = .8613$$

$\frac{2}{7}$ examples where $x_1 = 1$ and $\frac{5}{7}$ examples where $x_1 = 0$

$$H_{x_1=0} = -\frac{4}{5}log_2(\frac{4}{5}) - \frac{1}{5}log_2(\frac{1}{5}) = .7219$$

$$H_{x_1=1} = -\frac{1}{2}log_2(\frac{1}{2}) - \frac{1}{2}log_2(\frac{1}{2}) = 1$$

$$InfoGain_{x_1} = H_y - (\frac{5}{7}H_{x_1=0} + \frac{2}{7}H_{x_1=1}) = .0617$$

$\frac{4}{7}$ examples where $x_2 = 1$ and $\frac{3}{7}$ examples where $x_2 = 0$

$$H_{x_2=0} = -\frac{2}{3}log_2(\frac{2}{3}) - \frac{1}{3}log_2(\frac{1}{3}) = .9183$$

$$H_{x_2=1} = 0$$

$$InfoGain_{x_2} = H_y - (\frac{3}{7}H_{x_2=0} + \frac{4}{7}H_{x_2=1}) = .4696$$

$\frac{3}{7}$ examples where $x_3 = 1$ and $\frac{4}{7}$ examples where $x_3 = 0$

$$H_{x_3=0} = -\frac{3}{4}log_2(\frac{3}{4}) - \frac{1}{4}log_2(\frac{1}{4}) = .8113$$

$$H_{x_3=1} = -\frac{2}{3}log_2(\frac{2}{3}) - \frac{1}{3}log_2(\frac{1}{3}) = .9813$$

$$InfoGain_{x_3} = H_y - (\frac{4}{7}H_{x_3=0} + \frac{3}{7}H_{x_3=1}) = .0060$$

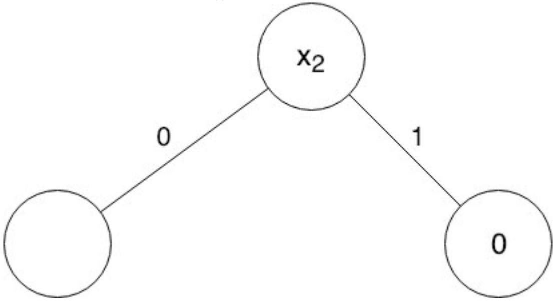
$\frac{3}{7}$ examples where $x_4 = 1$ and $\frac{4}{7}$ examples where $x_4 = 0$

$$H_{x_4=0} = 0$$

$$H_{x_4=1} = -\frac{2}{3}log_2(\frac{2}{3}) - \frac{1}{3}log_2(\frac{1}{3}) = .9813$$

$$InfoGain_{x_4} = H_y - (\frac{4}{7}H_{x_4=0} + \frac{3}{7}H_{x_4=1}) = .4696$$

We can either split on x_2 or x_4. Let's use x_2. This leads to the tree:



Now we need to decide what feature to split on when $x_2 = 0$.

x_1	x_3	x_4	y
0	1	0	0
0	1	1	1
1	0	1	1

Table 2: Training data for a Boolean classifier when $x_2 = 0$

$$H_y = -\frac{2}{3}log_2(\frac{2}{3}) - \frac{1}{3}log_2(\frac{1}{3}) = .9183$$

$\frac{1}{3}$ examples where $x_1 = 1$ and $\frac{2}{3}$ examples where $x_1 = 0$

$$H_{x_1=0} = -\frac{1}{2}log_2(\frac{1}{2}) - \frac{1}{2}log_2(\frac{1}{2}) = 1$$

$$H_{x_1=1} = 0$$

$$InfoGain_{x_1} = H_y - (\frac{5}{7}H_{x_1=0} + \frac{2}{7}H_{x_1=1}) = .2516$$

$\frac{2}{3}$ examples where $x_3 = 1$ and $\frac{1}{3}$ examples where $x_3 = 0$

$$H_{x_3=0} = 0$$

$$H_{x_3=1} = -\frac{1}{2}log_2(\frac{1}{2}) - \frac{1}{2}log_2(\frac{1}{2}) = 1$$

$$InfoGain_{x_3} = H_y - (\frac{1}{3}H_{x_3=0} + \frac{2}{3}H_{x_3=1}) = .2516$$

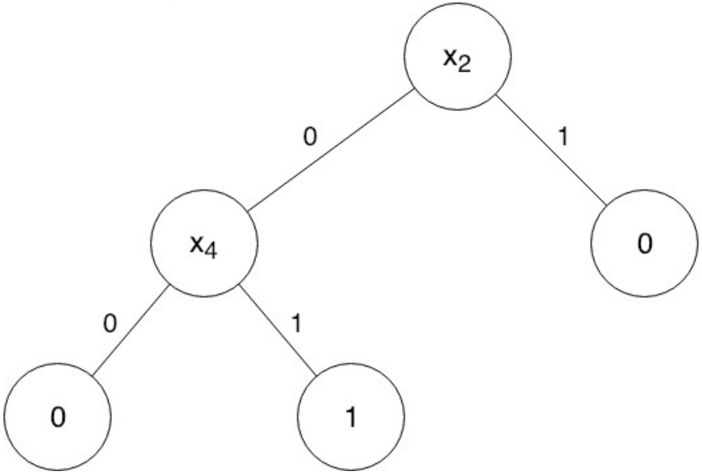
$\frac{2}{3}$ examples where $x_4 = 1$ and $\frac{1}{3}$ examples where $x_4 = 0$

$$H_{x_4=0} = 0$$

$$H_{x_4=1} = 0$$

$$InfoGain_{x_4} = H_y - (\frac{1}{3}H_{x_4=0} + \frac{2}{3}H_{x_4=1}) = .9183$$

We should split on x_4. This will lead to the tree:



(b) [2 points] Write the boolean function which your decision tree represents. Please use a table to describe the function — the columns are the input variables and label, i.e., x_1, x_2, x_3, x_4 and y; the rows are different input and function values.

x_1	x_2	x_3	x_4	y
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Table 3: Training data for a Boolean classifier

2. [17 points] Let us use a training dataset to learn a decision tree about whether to play tennis (**Page 39, Lecture: Decision Tree Learning**, accessible by clicking the link http://www.cs.utah.edu/~zhe/teach/pdf/decision-trees-learning.pdf). In the class, we have shown how to use information gain to construct the tree in ID3 framework.

   (a) [7 points] Now, please use majority error (ME) to calculate the gain, and select the best feature to split the data in ID3 framework. As in problem 1, please list every step in your tree construction, the attributes, how you calculate the gain of each attribute and how you split the dataset according to the selected attribute. Please also give a full structure of the tree.

$$ME(play) = \frac{5}{14}$$

$\frac{5}{14}$ examples where $O = S$, $\frac{4}{14}$ examples where $O = O$, and $\frac{5}{14}$ where $O = R$

$$ME(O = S) = \frac{2}{5}$$

$$ME(O = O) = 0$$

$$ME(O = R) = \frac{2}{5}$$

$$InfoGain(O) = ME(play) - (\frac{5}{14}ME(O = S) + \frac{4}{14}ME(O = O) + \frac{5}{14}ME(O = R)) = .0714$$

$\frac{4}{14}$ examples where $T = H$, $\frac{6}{14}$ examples where $T = M$, and $\frac{4}{14}$ where $T = C$

$$ME(T = H) = \frac{1}{2}$$

$$ME(T = M) = \frac{1}{3}$$

$$ME(T = C) = \frac{1}{4}$$

$$InfoGain(T) = ME(play) - (\frac{4}{14}ME(T = H) + \frac{6}{14}ME(T = M) + \frac{4}{14}ME(T = C)) = 0$$

$\frac{6}{14}$ examples where $H = H$ and $\frac{8}{14}$ where $T = N$

$$ME(H = H) = \frac{1}{2}$$

$$ME(H = N) = \frac{1}{4}$$

$$InfoGain(H) = ME(play) - (\frac{6}{14}ME(H = H) + \frac{8}{14}ME(H = N)) = 0$$

$\frac{8}{14}$ examples where $W = W$ and $\frac{6}{14}$ examples where $W = S$

$$ME(W = W) = \frac{1}{4}$$

$$ME(W = S) = \frac{1}{2}$$

$$InfoGain(W) = ME(play) - (\frac{8}{14}ME(W = W) + \frac{6}{14}ME(W = S)) = 0$$

We will split on outlook. This is the same feature that we split on when using entropy as the information gain. We now need to determine which features to split on when $O = S$ and $O = R$. Let's start with $O = S$.

$$ME(play) = \frac{2}{5}$$

$\frac{2}{5}$ examples where $T = H$, $\frac{2}{5}$ examples where $T = M$, and $\frac{1}{5}$ examples where $T = M$

$$ME(T = H) = 0$$

$$ME(T = M) = \frac{1}{2}$$

$$ME(T = C) = 0$$

$$InfoGain(T) = ME(play) - (\frac{2}{5}ME(T = H) + \frac{2}{5}ME(T = M) + \frac{1}{5}ME(T = C) = .2$$

$\frac{3}{5}$ examples where $H = H$ and $\frac{2}{5}$ examples where $H = N$

$$ME(H = H) = 0$$

$$ME(H = N) = 0$$

$$InfoGain(H) = .4$$

$\frac{3}{5}$ examples where $W = W$ and $\frac{2}{5}$ examples where $W = S$

$$ME(W = W) = \frac{1}{3}$$

$$ME(W = S) = \frac{1}{2}$$

$$InfoGain(W) = ME(play) - (\frac{3}{5}ME(W = W) + \frac{2}{5}ME(W = S)) = 0$$

We shall split on the humidity feature. This is the same feature we split on when we used entropy as information gain. Now let's see which feature we should split on when $O = R$.

$$ME(play) = \frac{2}{5}$$

$\frac{3}{5}$ examples where $T = M$ and $\frac{2}{5}$ examples where $T = C$

$$ME(T = M) = \frac{1}{3}$$

$$ME(T = C) = \frac{1}{2}$$

$$InfoGain(T) = ME(play) - (\frac{3}{5}ME(T = M) + \frac{2}{5}ME(T = C)) = 0$$

$\frac{1}{5}$ examples where $H = H$ and $\frac{4}{5}$ examples where $H = N$

$$ME(H = H) = 0$$

$$ME(H = N) = \frac{1}{2}$$

$$InfoGain(H) = ME(play) - (\frac{1}{5}ME(H = H) + \frac{4}{5}ME(H = N)) = 0$$

$\frac{3}{5}$ examples where $W = W$ and $\frac{2}{5}$ examples where $W = S$

$$ME(W = W) = 0$$

$$ME(W = S) = 0$$

$$InfoGain(W) = .4$$

Again this is the exact same feature we split on in class. We have the same tree as before.

(b) [7 points] Please use gini index (GI) to calculate the gain, and conduct tree learning with ID3 framework. List every step and the tree structure.

$$GI(play) = 1 - (\frac{9}{14}^2 + \frac{5}{14}^2) = .4592$$

$\frac{5}{14}$ examples where $O = S$, $\frac{4}{14}$ examples where $O = O$, and $\frac{5}{14}$ where $O = R$

$$GI(O = S) = 1 - (\frac{3}{5}^2 + \frac{2}{5}^2) = .48$$

$$GI(O = O) = 0$$

$$GI(O = R) = 1 - (\frac{3}{5}^2 + \frac{2}{5}^2) = .48$$

$$InfoGain(O) = GI(play) - (\frac{5}{14}GI(O = S) + \frac{5}{14}GI(O = R)) = .1163$$

$\frac{4}{14}$ examples where $T = H$, $\frac{6}{14}$ examples where $T = M$, and $\frac{4}{14}$ examples where $T = C$

$$GI(T = H) = 1 - (\frac{1}{2}^2 + \frac{1}{2}^2) = .5$$

$$GI(T = M) = 1 - (\frac{2}{3}^2 + \frac{1}{3}^2) = .4444$$

$$GI(T = C) = 1 - (\frac{3}{4}^2 + \frac{1}{4}^2) = .375$$

$$InfoGain(T) = GI(play) - (\frac{4}{14}GI(T = H) + \frac{6}{14}GI(T = M) + \frac{4}{14}GI(T = C)) = .0187$$

$\frac{7}{14}$ examples where $H = H$ and $\frac{7}{14}$ examples where $H = N$

$$GI(H = H) = 1 - (\frac{4}{7}^2 + \frac{3}{7}^2) = .4898$$

$$GI(H=N) = 1 - (\frac{6}{7}^2 + \frac{1}{7}^2) = .2449$$

$$InfoGain(H) = GI(play) - (\frac{7}{14}GI(H=H) + \frac{7}{14}GI(H=N)) = .0918$$

$\frac{8}{14}$ examples where $W = W$ and $\frac{6}{14}$ examples where $W = S$

$$GI(W=W) = 1 - (\frac{1}{4}^2 + \frac{3}{4}^2) = .375$$

$$GI(W=S) = 1 - (\frac{1}{2}^2 + \frac{1}{2}^2) = .5$$

$$InfoGain(W) = GI(play) - (\frac{8}{14}GI(W=W) + \frac{6}{14}GI(W=S)) = .0306$$

Again we will split on the outlook feature just like from the lecture. Let's look at $O = S$.

$$GI(play) = 1 - (\frac{3}{5}^2 + \frac{2}{5}^2) = .48$$

$\frac{2}{5}$ examples where $T = H$, $\frac{2}{5}$ examples where $T = M$, and $\frac{1}{5}$ examples where $T = C$

$$GI(T=H) = 0$$

$$GI(T=M) = 1 - (\frac{1}{2}^2 + \frac{1}{2}^2) = .5$$

$$GI(T=C) = 0$$

$$InfoGain(T) = \frac{2}{5}GI(T=M) = .2$$

$\frac{3}{5}$ examples where $H = H$ and $\frac{2}{5}$ examples where $H = N$

$$GI(H=H) = 0$$

$$GI(H=N) = 0$$

$$InfoGain(H) = GI(play) = .48$$

$\frac{3}{5}$ examples where $W = W$ and $\frac{2}{5}$ examples where $W = S$

$$GI(W=W) = 1 - (\frac{2}{3}^2 + \frac{1}{3}^2) = .4444$$

$$GI(W=S) = 1 - (\frac{1}{2}^2 + \frac{1}{2}^2) = .5$$

$$InfoGain(W) = GI(play) - (\frac{3}{5}GI(W=W) + \frac{2}{5}GI(W=S)) = .0133$$

We split on the humidity feature. This is the same as the tree in class. Let's see which feature to split on when $O = R$

$$GI(play) = 1 - (\frac{3}{5}^2 + \frac{2}{5}^2) = .48$$

$\frac{3}{5}$ examples where $T = M$ and $\frac{2}{5}$ examples where $T = C$

$$GI(T = M) = 1 - (\frac{1}{3}^2 + \frac{2}{3}^2) = .4444$$

$$GI(T = C) = 1 - (\frac{1}{2}^2 + \frac{1}{2}^2) = .5$$

$$InfoGain(T) = GI(play) - (\frac{3}{5}GI(T = M) + \frac{2}{5}GI(T = C)) = .0133$$

$\frac{2}{5}$ examples where $H = H$ and $\frac{3}{5}$ examples where $H = N$

$$GI(H = H) = 1 - (\frac{1}{2}^2 + \frac{1}{2}^2) = .5$$

$$GI(H = N) = 1 - (\frac{2}{3}^2 + \frac{1}{3}^2) = .4444$$

$$InfoGain(H) = GI(play) - (\frac{2}{5}GI(H = H) + \frac{3}{5}GI(H = N)) = .0133$$

$\frac{3}{5}$ examples where $W = W$ and $\frac{2}{5}$ examples where $W = S$

$$GI(W = W) = 0$$

$$GI(W = S) = 0$$

$$InfoGain(W) = GI(play) = .48$$

We will split on the weather feature again like before. The tree looks the exact same.

(c) [3 points] Compare the two trees you just created with the one we built in the class (see Page 58 of the lecture slides). Are there any differences? Why?

There are no differences between any of the trees. This is because all the variants of information gain probably act the same especially for smaller data sets.

3. [16 points] Continue with the same training data in Problem 2. Suppose before the tree construction, we receive one more training instance where Outlook's value is missing: {Outlook: Missing, Temperature: Mild, Humidity: Normal, Wind: Weak, Play: Yes}.

(a) [3 points] Use the most common value in the training data as the missing value, and calculate the information gains of the four features. Indicate the best feature.

We could either use sunny or rainy as the most common value because they both occupy $\frac{5}{14}$ examples. Let's use sunny.

$$H(play) = -\frac{10}{15}log(\frac{10}{15}) - \frac{5}{15}log(\frac{5}{15}) = .9183$$

$\frac{6}{15}$ examples where $O = S$, $\frac{4}{15}$ examples where $O = O$, and $\frac{5}{15}$ examples where $O = R$

$$H(O = S) = -\frac{1}{2}log(\frac{1}{2}) - \frac{1}{2}log(\frac{1}{2}) = 1$$

$$H(O = O) = 0$$

$$H(O = R) = -\frac{3}{5}log(\frac{3}{5}) - \frac{2}{5}log(\frac{2}{5}) = .9710$$

$$InfoGain(O) = H(play) - (\frac{6}{15}H(O = S)\frac{5}{15}H(O = R)) = .1946$$

$\frac{4}{15}$ examples where $T = H$, $\frac{7}{15}$ examples where $T = M$, and $\frac{4}{15}$ where $T = C$

$$H(T = H) = -\frac{1}{2}log(\frac{1}{2}) - \frac{1}{2}log(\frac{1}{2}) = 1$$

$$H(T = M) = -\frac{5}{7}log(\frac{5}{7}) - \frac{2}{7}log(\frac{2}{7}) = .8631$$

$$H(T = C) = -\frac{3}{4}log(\frac{3}{4}) - \frac{1}{4}log(\frac{1}{4}) = .8113$$

$$InfoGain(T) = H(play) - (\frac{4}{15}H(T = H) + \frac{7}{15}H(T = M) + \frac{4}{15}H(T = C)) = .0325$$

$\frac{7}{15}$ where $H = H$ and $\frac{8}{15}$ examples where $H = N$

$$H(H = H) = -\frac{3}{7}log(\frac{3}{7}) - \frac{4}{7}log(\frac{4}{7}) = .9852$$

$$H(H = N) = -\frac{7}{8}log(\frac{7}{8}) - \frac{1}{8}log(\frac{1}{8}) = .5436$$

$$InfoGain(H) = H(play) - (\frac{7}{15}H(H = H) + \frac{8}{15}H(H = N)) = .1686$$

$\frac{9}{15}$ examples where $W = W$ and $\frac{6}{15}$ examples where $W = S$

$$H(W = W) = -\frac{2}{9}log(\frac{2}{9}) - \frac{7}{9}log(\frac{7}{9}) = .7642$$

$$H(W = S) = -\frac{1}{2}log(\frac{1}{2}) - \frac{1}{2}log(\frac{1}{2}) = 1$$

$$InfoGain(W) = H(play) - (\frac{9}{15}H(W = W) + \frac{6}{15}H(W = S)) = .0598$$

The best feature is outlook.

(b) [3 points] Use the most common value among the training instances with the same label, namely, their attribute "Play" is "Yes", and calculate the information gains of the four features. Indicate the best feature

Among the training instances with the same label the most common value was

$O = O$. $\frac{5}{15}$ examples where $O = S$, $\frac{5}{15}$ examples where $O = O$, and $\frac{5}{15}$ examples where $O = R$

$$H(O = S) = -\frac{3}{5}log(\frac{3}{5}) - \frac{2}{5}log(\frac{2}{5}) = .9710$$

$$H(O = O) = 0$$

$$H(O = R) = -\frac{3}{5}log(\frac{3}{5}) - \frac{2}{5}log(\frac{2}{5}) = .9710$$

$$InfoGain(O) = H(play) - (\frac{5}{15}H(O = S) + \frac{5}{15}H(O = R)) = .2710$$

The best feature to split on is outlook.

(c) [3 points] Use the fractional counts to infer the feature values, and then calculate the information gains of the four features. Indicate the best feature.

$\frac{5}{14}$ examples where $O = S$, $\frac{4}{14}$ examples where $O = O$, and $\frac{5}{14}$ examples where $O = R$

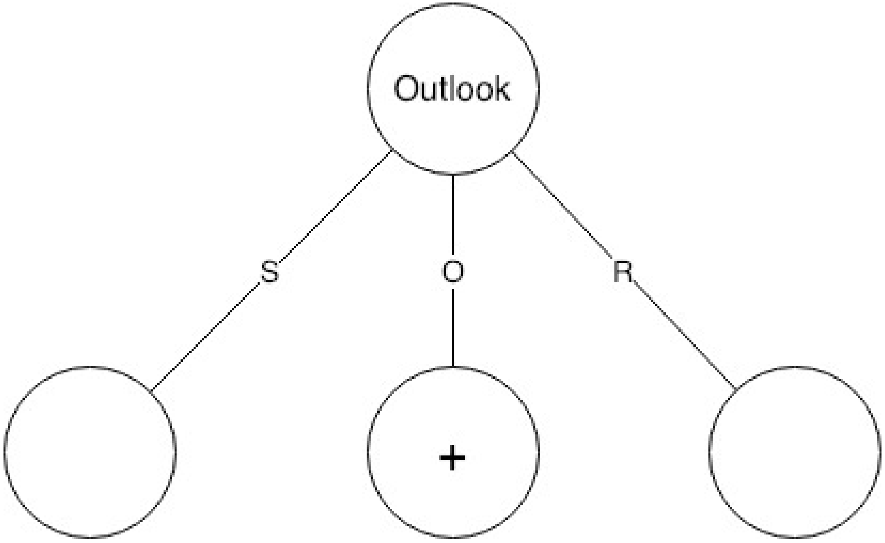
$$H(O = S) = -\frac{3}{5 + \frac{5}{14}}log(\frac{3}{5 + \frac{5}{14}}) - \frac{2 + \frac{5}{14}}{5 + \frac{5}{14}}log(\frac{2 + \frac{5}{14}}{5 + \frac{5}{14}}) = .9896$$

$$H(O = O) = 0$$

$$H(O = R) = -\frac{3}{5 + \frac{5}{14}}log(\frac{3}{5 + \frac{5}{14}}) - \frac{2 + \frac{5}{14}}{5 + \frac{5}{14}}log(\frac{2 + \frac{5}{14}}{5 + \frac{5}{14}}) = .9896$$

$$InfoGain(O) = H(play) - (\frac{5}{14}H(O = S) + \frac{5}{14}H(O = R)) = .2114$$

We split on outlook. Here's what our tree looks like so far:



(d) [7 points] Continue with the fractional examples, and build the whole tree with information gain. List every step and the final tree structure.

Let's see which feature to split on when $O = S$.

$$H(play) = -\frac{3}{5+\frac{5}{14}}log(\frac{3}{5+\frac{5}{14}}) - \frac{2+\frac{5}{14}}{5+\frac{5}{14}}log(\frac{2+\frac{5}{14}}{5+\frac{5}{14}}) = .9896$$

$\frac{2}{5+\frac{5}{14}}$ examples where $T = H$, $\frac{2+\frac{5}{14}}{5+\frac{5}{14}}$ examples where $T = M$, and $\frac{1}{5+\frac{5}{14}}$ examples where $T = C$.

$$H(T = H) = 0$$

$$H(T = M) = -\frac{1}{2+\frac{5}{14}}log(\frac{1}{2+\frac{5}{14}}) - \frac{1+\frac{5}{14}}{2+\frac{5}{14}}log(\frac{1+\frac{5}{14}}{2+\frac{5}{14}}) = .8315$$

$$H(T = C) = 0$$

$$InfoGain(T) = Hplay - \frac{2+\frac{5}{14}}{5+\frac{5}{14}}H(T = M) = .6237$$

$\frac{3}{5+\frac{5}{14}}$ examples where $H = H$ and $\frac{2+\frac{5}{14}}{5+\frac{5}{14}}$ examples where $H = N$

$$H(H = H) = 0$$

$$H(H = N) = 0$$

$$InfoGain(H) = H(play) = .9896$$

$\frac{2+\frac{5}{14}}{5+\frac{5}{14}}$ examples where $W = W$ and $\frac{3}{5+\frac{5}{14}}$ examples where $W = S$

$$H(W = W) = -\frac{1}{2+\frac{5}{14}}log(\frac{1}{2+\frac{5}{14}}) - \frac{1+\frac{5}{14}}{2+\frac{5}{14}}log(\frac{1+\frac{5}{14}}{2+\frac{5}{14}}) = .8315$$

$$H(W = S) = -\frac{1}{2}log(\frac{1}{2}) - \frac{1}{2}log(\frac{1}{2}) = 1$$

$$InfoGain(W) = H(play) - (\frac{2+\frac{5}{14}}{5+\frac{5}{14}}H(W = W) + \frac{3}{5+\frac{5}{14}}) = .0637$$

We will split on the humidity feature. Now let's see which feature to split on when $O = R$.

$$H(play) = -\frac{2}{5+\frac{5}{14}}log(\frac{2}{5+\frac{5}{14}}) - \frac{3+\frac{5}{14}}{5+\frac{5}{14}}log(\frac{3+\frac{5}{14}}{5+\frac{5}{14}}) = .9532$$

$\frac{3+\frac{5}{14}}{5+\frac{5}{14}}$ examples where $T = M$, and $\frac{2}{5+\frac{5}{14}}$ examples where $T = C$.

$$H(T = M) = -\frac{1}{3+\frac{5}{14}}log(\frac{1}{3+\frac{5}{14}}) - \frac{2+\frac{5}{14}}{3+\frac{5}{14}}log(\frac{2+\frac{5}{14}}{3+\frac{5}{14}}) = .8787$$

$$H(T = C) = 1$$

$$InfoGain(T) = H(play) - (\frac{3+\frac{5}{14}}{5+\frac{5}{14}}H(T=M) + \frac{2}{5+\frac{5}{14}}) = .0292$$

$\frac{2}{5+\frac{5}{14}}$ examples where $H = H$, and $\frac{3+\frac{5}{14}}{5+\frac{5}{14}}$ examples where $H = N$.

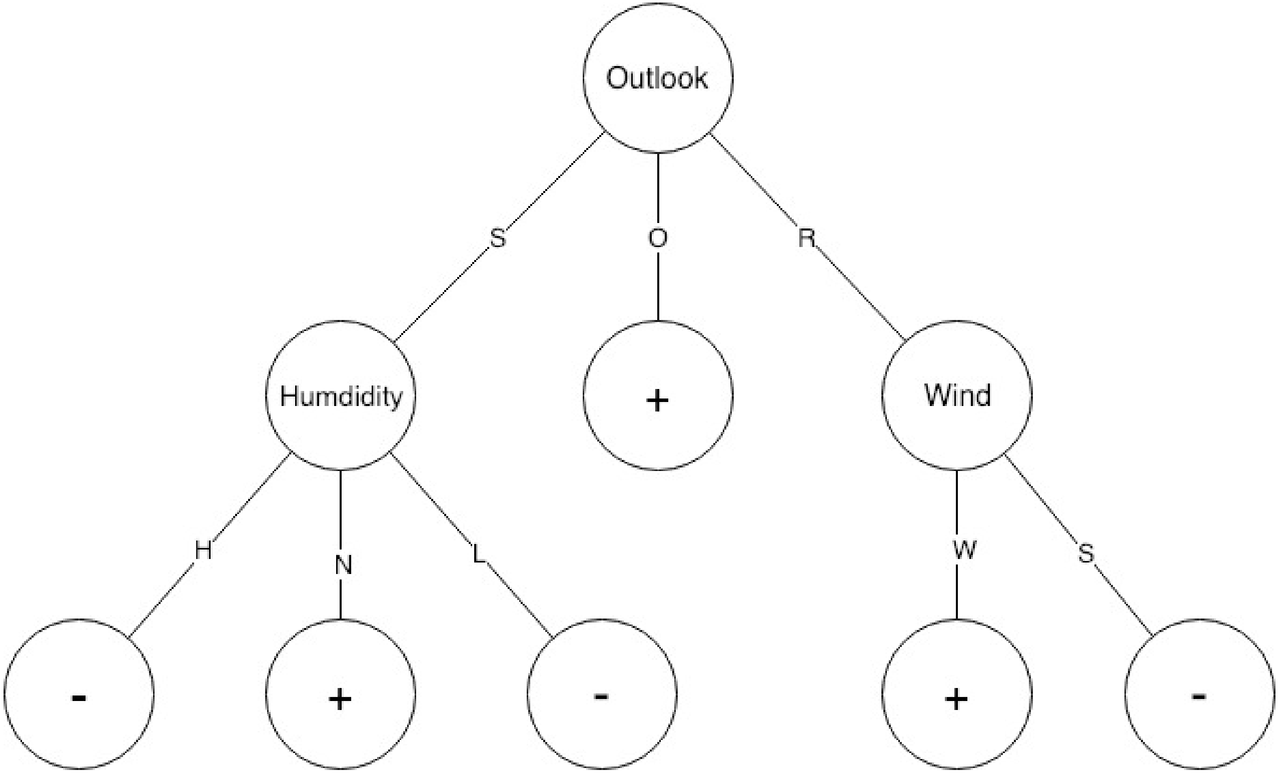
$$H(H=H) = -\frac{1}{2}log(\frac{1}{2}) - \frac{1}{2}log(\frac{1}{2}) = 1$$

$$H(H=N) = -\frac{2+\frac{5}{14}}{3+\frac{5}{14}}log(\frac{2+\frac{5}{14}}{3+\frac{5}{14}}) - \frac{1}{3+\frac{5}{14}}log(\frac{1}{3+\frac{5}{14}}) = .8787$$

$$InfoGain(H) = H(play) - (\frac{2}{5+\frac{5}{14}}H(H=H) + \frac{3+\frac{5}{14}}{5+\frac{5}{14}}H(H=N)) = .0292$$

$$InfoGain(W) = H(play) = .9532$$

We split on the wind feature.



4. [**Bonus question 1**] [5 points]. Prove that the information gain is always non-negative. That means, as long as we split the data, the purity will never get worse! (Hint: use convexity)

5. [**Bonus question 2**] [5 points]. We have discussed how to use decision tree for regression (i.e., predict numerical values) — on the leaf node, we simply use the average of the (numerical) labels as the prediction. Now, to construct a regression tree, can you invent a gain to select the best attribute to split data in ID3 framework?

2 Decision Tree Practice [60 points]

1. [5 Points] Starting from this assignment, we will build a light-weighted machine learning library. To this end, you will first need to create a code repository in Github.com. Please refer to the short introduction in the appendix and the official tutorial to create an account and repository. Please commit a README.md file in your repository, and write one sentence: ”This is a machine learning library developed by **Your Name** for CS5350/6350 in University of Utah”. You can now create a first folder, ”DecisionTree”. Please leave the link to your repository in the homework submission. We will check if you have successfully created it.
   https://github.com/nrtominaga/MachineLearning

2. [30 points] We will implement a decision tree learning algorithm for car evaluation task. The dataset is from UCI repository(`https://archive.ics.uci.edu/ml/datasets/car+evaluation`). Please download the processed dataset (car.zip) from Canvas. In this task, we have 6 car attributes, and the label is the evaluation of the car. The attribute and label values are listed in the file “data-desc.txt”. All the attributes are categorical. The training data are stored in the file “train.csv”, consisting of $1,000$ examples. The test data are stored in “test.csv”, and comprise 728 examples. In both training and test datasets, attribute values are separated by commas; the file “data-desc.txt” lists the attribute names in each column.

   Note: we highly recommend you to use Python for implementation, because it is very convenient to load the data and handle strings. For example, the following snippet reads the CSV file line by line and split the values of the attributes and the label into a list, “terms”. You can also use “dictionary” to store the categorical attribute values. In the web are numerous tutorials and examples for Python. if you have issues, just google it!

```
with open(CSVfile, ’r’) as f:
    for line in f:
        terms = line.strip().split(’,’)
        process one training example
```

   (a) [15 points] Implement the ID3 algorithm that supports, information gain, majority error and gini index to select attributes for data splits. Besides, your ID3 should allow users to set the maximum tree depth. Note: you do not need to convert categorical attributes into binary ones and your tree can be wide here.

   (b) [10 points] Use your implemented algorithm to learn decision trees from the training data. Vary the maximum tree depth from 1 to 6 — for each setting, run your algorithm to learn a decision tree, and use the tree to predict both the training and test examples. Note that if your tree cannot grow up to 6 levels, then you can stop at the maximum level. Report in a table the average prediction errors on each dataset when you use information gain, majority error and gini index heuristics, respectively.

	entropy	*ME*	*GI*
1	69.80	69.80	69.80
2	77.80	70.20	77.80
3	81.90	77.10	82.40
4	91.80	85.70	91.10
5	97.30	95.40	97.30
6	100.00	100.00	100.00

Table 4: Training data

	entropy	*ME*	*GI*
1	70.33	70.33	70.33
2	77.75	67.58	77.75
3	80.36	75.96	81.59
4	85.30	83.10	86.68
5	91.62	88.60	91.62
6	91.62	88.60	91.62

Table 5: Test data

(c) [5 points] What can you conclude by comparing the training errors and the test errors?

Overfitting does occur but it's not that bad. Seems like entropy and gini index are pretty good heuristics.

3. [25 points] Next, modify your implementation a little bit to support numerical attributes. We will use a simple approach to convert a numerical feature to a binary one. We choose the media (NOT the average) of the attribute values (in the training set) as the threshold, and examine if the feature is bigger (or less) than the threshold. We will use another real dataset from UCI repository(`https://archive.ics.uci.edu/ml/datasets/Bank+Marketing`). This dataset contains 16 attributes, including both numerical and categorical ones. Please download the processed dataset from Canvas (bank.zip). The attribute and label values are listed in the file "data-desc.txt". The training set is the file "train.csv", consisting of $5,000$ examples, and the test "test.csv" with $5,000$ examples as well. In both training and test datasets, attribute values are separated by commas; the file "data-desc.txt" lists the attribute names in each column.

(a) [10 points] Let us consider "unkown" as a particular attribute value, and hence we do not have any missing attributes for both training and test. Vary the maximum tree depth from 1 to 16 — for each setting, run your algorithm to learn a decision tree, and use the tree to predict both the training and test examples. Again, if your tree cannot grow up to 16 levels, stop at the maximum level. Report in a table the average prediction errors on each dataset when you use information gain, majority error and gini index heuristics, respectively.

	$entropy$	ME	GI
1	88.08	89.12	89.12
2	89.40	89.58	89.58
3	89.94	90.32	90.64
4	92.00	91.26	92.48
5	93.76	92.16	93.92
6	95.22	92.66	95.16
7	96.26	92.98	96.30
8	97.08	93.76	97.30
9	97.72	94.96	97.82
10	98.20	95.82	98.22
11	98.48	96.48	98.52
12	98.62	97.16	98.60
13	98.68	97.74	98.68
14	98.68	98.10	98.68
15	98.68	98.40	98.68
16	98.68	98.68	98.68

Table 6: Training data w/ unkowns

(b) [10 points] Let us consider "unkown" as attribute value missing. Here we simply complete it with the majority of other values of the same attribute in the training set. Vary the maximum tree depth from 1 to 16 — for each setting, run your algorithm to learn a decision tree, and use the tree to predict both the training and test examples. Report in a table the average prediction errors on each dataset when you use information gain, majority error and gini index heuristics, respectively.

(c) [5 points] What can you conclude by comparing the training errors and the test errors, with different tree depths, as well as different ways to deal with "unkown" attribute values?

Overfitting is much more pronounced here. Replacing the unknowns helps a little. Not too much difference between the heuristics.

	entropy	*ME*	*GI*
1	87.52	88.34	88.34
2	88.86	89.12	89.12
3	89.26	89.06	88.78
4	88.52	88.76	88.30
5	87.40	88.66	87.26
6	86.42	88.34	86.18
7	85.96	88.32	85.32
8	85.40	87.92	85.12
9	85.00	87.20	84.74
10	84.56	86.70	84.48
11	84.50	86.50	84.16
12	84.08	85.98	83.94
13	84.10	85.74	83.96
14	84.10	85.30	83.96
15	84.10	84.92	83.96
16	84.10	84.74	83.96

Table 7: Test data w/ unkowns

	entropy	*ME*	*GI*
1	88.08	89.12	89.12
2	89.40	89.52	89.40
3	89.86	90.16	90.12
4	91.76	91.08	91.96
5	93.30	92.02	93.06
6	94.60	92.48	94.50
7	95.88	92.86	95.90
8	96.60	93.66	96.74
9	97.34	94.78	97.38
10	97.88	95.54	97.92
11	98.16	96.32	98.20
12	98.30	97.12	98.30
13	98.46	97.62	98.46
14	98.48	97.94	98.48
15	98.48	98.22	98.48
16	98.48	98.48	98.48

Table 8: Training data w/o unkowns

	$entropy$	ME	GI
1	87.52	88.34	88.34
2	88.86	88.92	88.86
3	89.44	89.18	89.08
4	88.80	88.80	88.86
5	88.30	88.90	88.36
6	86.56	88.66	86.94
7	85.76	88.54	85.72
8	85.32	88.06	85.04
9	84.66	87.20	84.74
10	84.72	86.94	84.60
11	84.34	86.14	84.10
12	84.36	85.66	84.10
13	84.30	85.18	84.12
14	84.20	84.98	84.02
15	84.18	84.60	84.00
16	84.18	84.40	84.00

Table 9: Test data w/o unkowns