# CS 5350/6350: Machine Learning Fall 2016

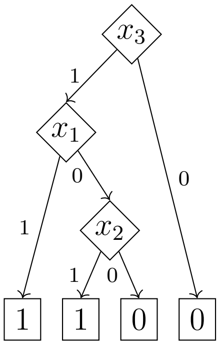
Homework 1

Gopal Menon

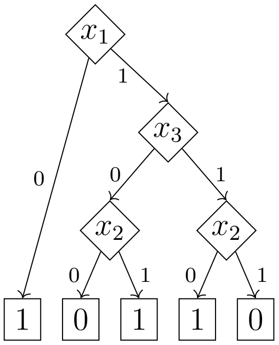
September 13, 2016

## 1 Decision trees (35 points)

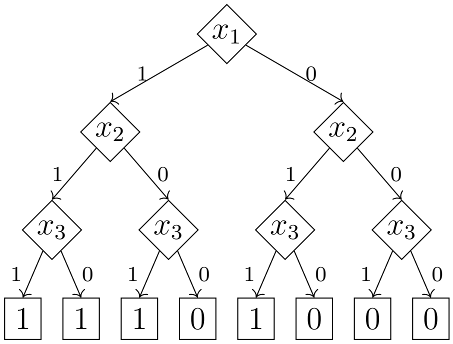
1. (a) $(x_1 \vee x_2) \wedge x_3$



(b) $(x_1 \wedge x_2)$ xor $(\neg x_1 \vee x_3)$



(c) The 2-of-3 function defined as follows: at least 2 of $\{x_1, x_2, x_3\}$ should be true for the output to be true.



2. (a) [2 points] How many possible functions are there to map these four features to a Boolean decision?

   The number of possible rows in a truth table with all possible values of the four features will be

   Number of types of Berries × Number of types of Balls × Number of colors × Number of Pokémon types
   $= 2 \times 3 \times 3 \times 4 = 72$

   Each row in this truth table can have a value of $Yes$ or $No$ as a label for whether the Pokémon can be caught or not. The $Yes$ and $No$ can be represented as bits 1 and 0. Each possible function will map to one combination of these labels and will be of the form of a 72-bit binary number. So the total possible number of functions will be $2^{72}$, which is the count of all the possible 72-bit binary numbers.

   (b) [2 points] What is the entropy of the labels in this data? (When calculating entropy, The base of the logarithm should be 2.)

   The entropy of a collection $S$ where the target label can take on $c$ different values is defined as [1]

$$Entropy(S) = \sum_{i=1}^{c} -p_i log_2 p_i$$

   where $p_i$ is the proportion of $S$ belonging to label $i$.

$$\begin{aligned} Entropy(Pok\acute{e}monData) &= -\frac{8}{16} log_2 \frac{8}{16} - \frac{8}{16} log_2 \frac{8}{16} \\ &= -\frac{1}{2} log_2 \frac{1}{2} - \frac{1}{2} log_2 \frac{1}{2} \\ &= -log_2 \frac{1}{2} \\ &= 1 \end{aligned}$$

   (c) [8 points] Calculate information gain for four features respectively. Keep 3 significant digits.

$$
\begin{aligned}
Values(Berry) &= Yes, No \\
S &= [8+, 8-] \\
S_{Yes} &= [6+, 1-] \\
S_{No} &= [2+, 7-] \\
Gain(S, Berry) &= Entropy(S) - \frac{7}{16} Entropy(S_{Yes}) - \frac{9}{16} Entropy(S_{No}) \\
&= 1 - \frac{7}{16} \times 0.5917 - \frac{9}{16} \times 0.7642 \\
&= 0.311
\end{aligned}
$$

$$
\begin{aligned}
Values(Ball) &= Poké, Great, Ultra \\
S_{Poké} &= [1+, 5-] \\
S_{Great} &= [4+, 3-] \\
S_{Ultra} &= [3+, 0-] \\
Gain(S, Ball) &= Entropy(S) - \frac{6}{16} Entropy(S_{Poké}) - \frac{7}{16} Entropy(S_{Great}) - \frac{3}{16} Entropy(S_{Ultra}) \\
&= 1 - \frac{6}{16} \times 0.65 - \frac{7}{16} \times 0.9852 - \frac{3}{16} \times 0 \\
&= 0.3252
\end{aligned}
$$

$$
\begin{aligned}
Values(Color) &= Green, Yellow, Red \\
S_{Green} &= [2+, 1-] \\
S_{Yellow} &= [3+, 4-] \\
S_{Red} &= [3+, 3-] \\
Gain(S, Color) &= Entropy(S) - \frac{3}{16} Entropy(S_{Green}) - \frac{7}{16} Entropy(S_{Yellow}) \\
&- \frac{6}{16} Entropy(S_{Red}) \\
&= 1 - \frac{3}{16} \times 0.9183 - \frac{7}{16} \times 0.9852 - \frac{6}{16} \times 1 \\
&= 0.0218
\end{aligned}
$$

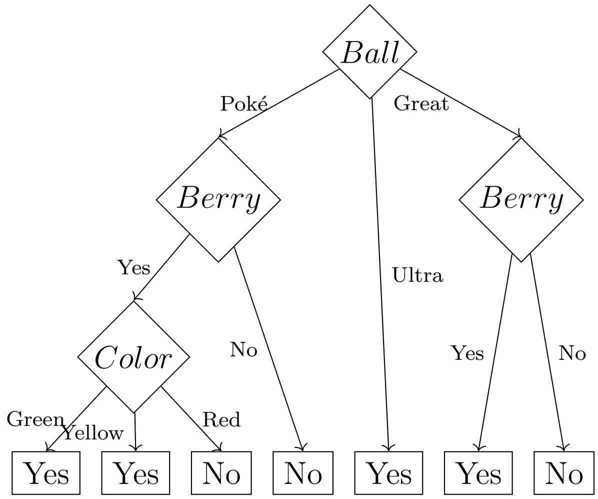
$$
\begin{aligned}
Values(Type) &= Normal, Water, Flying, Psychic \\
S_{Normal} &= [3+, 3-] \\
S_{Water} &= [2+, 2-] \\
S_{Flying} &= [3+, 1-] \\
S_{Psychic} &= [0+, 2-]
\end{aligned}
$$

$$\begin{aligned}Gain(S, Type) &= Entropy(S) - \frac{6}{16}Entropy(S_{Normal}) - \frac{4}{16}Entropy(S_{Water}) \\ &- \frac{4}{16}Entropy(S_{Flying}) - \frac{2}{16}Entropy(S_{Psychic}) \\ &= 1 - \frac{6}{16} \times 1 - \frac{4}{16} \times 1 - \frac{4}{16} \times 0.8113 - \frac{2}{16} \times 0 \\ &= 0.172\end{aligned}$$

(d) [3 points] According to your results, using ID3 algorithm which attribute should be root for the decision tree?

The root should be feature *Ball*, since it has the largest entropy gain.

(e) [4 points] Construct a decision with the root you selected in the previous question. You do not have to use the ID3 algorithm here, you can show any tree with the chosen root.



(f) [2 points] Using your decision tree to predict label in the test set in the table below, what is your label for the each example? What is your accuracy?

Only one out of three was classified correctly. So accuracy is low.

| Berry | Ball | Color | Type | Caught | Prediction |
|---|---|---|---|---|---|
| Yes | Great | Yellow | Psychic | Yes | Yes |
| Yes | Poké | Green | Flying | No | Yes |
| No | Ultra | Red | Water | No | Yes |

(g) [1 points] Do you think it is a good idea to use decision tree in this Pokémon Go problem?

It is not a good idea to use a decision tree for this particular Pokémon Go problem. The training data looks to be not enough for a learning algorithm and the test data seems to be adversarial. A decision tree may perform better with a larger test data set.

3. Recall that in the ID3 algorithm, we want to identify the best attribute that splits the examples that are relatively pure in one label. Apart from entropy, which you used in

the previous question, there are other methods to measure impurity. One such impurity measure is the Gini measure, that is used in the CART family of algorithms. If there are $k$ possible outcomes $1, \cdots, i, \cdots, k$, each with a probability $p_1, \cdots, p_i, \cdots, p_k$ of occurring, the Gini measure is defined as:

$$Gini(p_1, \cdots, p_k) = 1 - \sum_{i=1}^{k} p_i^2$$

The Gini measure can be used to replace entropy in the definition of information gain to pick the best attribute.

(a) [4 points] Using the Gini measure, calculate the information gain for the four features respectively. Use 3 significant digits.

$$\begin{aligned}
Gini(Pok\acute{e}monData) &= 1 - \left(\frac{8}{16}\right)^2 - \left(\frac{8}{16}\right)^2 \\
&= 1 - 0.25 - 0.25 \\
&= 0.5 \\
Gain(S, Berry) &= Gini(S) - \frac{7}{16}Gini(S_{Yes}) - \frac{9}{16}Gini(S_{No}) \\
&= 0.5 - \frac{7}{16}\left(1 - \left(\frac{6}{7}\right)^2 - \left(\frac{1}{7}\right)^2\right) - \frac{9}{16}\left(1 - \left(\frac{2}{9}\right)^2 - \left(\frac{7}{9}\right)^2\right) \\
&= 0.5 - \frac{7}{16} \times \frac{12}{49} - \frac{9}{16} \times \frac{28}{81} \\
&= 0.5 - 0.1071 - 0.1944 \\
&= 0.198
\end{aligned}$$

$$\begin{aligned}
Gain(S, Ball) &= Gini(S) - \frac{6}{16}Gini(S_{Pok\acute{e}}) - \frac{7}{16}Gini(S_{Great}) - \frac{3}{16}Gini(S_{Ultra}) \\
&= 0.5 - \frac{6}{16}\left(1 - \left(\frac{1}{6}\right)^2 - \left(\frac{5}{6}\right)^2\right) - \frac{7}{16}\left(1 - \left(\frac{4}{7}\right)^2 - \left(\frac{3}{7}\right)^2\right) \\
&- \frac{3}{16}\left(1 - \left(\frac{3}{3}\right)^2 - \left(\frac{0}{3}\right)^2\right) \\
&= 0.5 - \frac{6}{16} \times \frac{10}{36} - \frac{7}{16} \times \frac{24}{49} - \frac{3}{16} \times 0 \\
&= 0.5 - 0.1042 - 0.2143 \\
&= 0.181
\end{aligned}$$

$$
\begin{aligned}
Gain(S, Color) &= Gini(S) - \frac{3}{16}Gini(S_{Green}) - \frac{7}{16}Gini(S_{Yellow}) - \frac{6}{16}Gini(S_{Red}) \\
&= 0.5 - \frac{3}{16}\left(1 - \left(\frac{2}{3}\right)^2 - \left(\frac{1}{3}\right)^2\right) - \frac{7}{16}\left(1 - \left(\frac{3}{7}\right)^2 - \left(\frac{4}{7}\right)^2\right) \\
&- \frac{6}{16}\left(1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2\right) \\
&= 0.5 - \frac{3}{16} \times \frac{4}{9} - \frac{7}{16} \times \frac{24}{49} - \frac{6}{16} \times \frac{1}{2} \\
&= 0.5 - 0.0833 - 0.2143 - 0.1875 \\
&= 0.015
\end{aligned}
$$

$$
\begin{aligned}
Gain(S, Type) &= Gini(S) - \frac{6}{16}Gini(S_{Normal}) - \frac{4}{16}Gini(S_{Water}) - \frac{4}{16}Gini(S_{Flying}) \\
&- \frac{2}{16}Gini(S_{Psychic}) \\
&= 0.5 - \frac{6}{16}\left(1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2\right) - \frac{4}{16}\left(1 - \left(\frac{2}{4}\right)^2 - \left(\frac{2}{4}\right)^2\right) \\
&- \frac{4}{16}\left(1 - \left(\frac{3}{4}\right)^2 - \left(\frac{1}{4}\right)^2\right) - \frac{2}{16}\left(1 - \left(\frac{0}{2}\right)^2 - \left(\frac{2}{2}\right)^2\right) \\
&= 0.5 - \frac{6}{16} \times \frac{1}{2} - \frac{4}{16} \times \frac{1}{2} - \frac{4}{16} \times \frac{6}{16} - \frac{2}{16} \times 0 \\
&= 0.5 - 0.1875 - 0.125 - 0.09375 \\
&= 0.094
\end{aligned}
$$

(b) [3 points] According to your results in the last question, which attribute should be the root for the decision tree? Do these two measures (entropy and Gini) lead to the same tree?

As per the results in the last question based on Gini calculation, the attribute *Berry* must be the root for the decision tree as it has the maximum gain. In the case of Entropy calculation, the attribute *Ball* was the root of the decision tree. So the two measures will lead to different trees.

## 2 Linear Classifiers (15 points)

In the questions in this section, we have four features $x_1$, $x_2$, $x_3$ and $x_4$ and the label is represented by $o$.

1. [3 points] Write a linear classifier that correctly classifies the given dataset. You don't need to run any learning algorithm here. Try to find the weights and the bias of the classifier using the definition of linear separators.

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $o$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | -1 |

I obtained the weights and bias by trial and error.

$$w = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

$$b = -1$$

2. [5 points] Suppose the dataset below is an extension of the above dataset. Check if your classifier from the previous question correctly classifies the dataset. Report its accuracy.

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $o$ |
|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | -1 |

Using the above classifier, the output is

| *Classifier Output* | *Expected Output* |
|---|---|
| 1 | 1 |
| 1 | 1 |
| 1 | 1 |
| -1 | 1 |
| 1 | 1 |
| 1 | 1 |
| 1 | -1 |

The classifier correctly classified 5 out of 7 outputs, giving an accuracy of 71.43%.

3. [7 points] Given the remaining missing data points of the above dataset in the table below, find a linear classifier that correctly classifies the whole dataset (all three tables together).

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $o$ |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | -1 |
| 0 | 1 | 1 | 0 | -1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 |

Upon writing down the entire dataset, it becomes evident that the output is 1 when $x_1 = 1$. When $x_1 = 0$, the output is $-1$, when $x_4 = 0$ and 1 when $x_4 = 1$. The weight and bias will be

$$w = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$b = -1$$

## 3 Experiments

### Setting A [25 points]

1. [10 points] **Implementation**

   (a) [4 points] I used the Pokémon training and test data to verify that the decision tree had been coded properly. The decision tree was coded based on the algorithm in Tom Mitchellś book [1]. A random number generator was used when I needed to break a tie in the selection of the most common value of an attribute. To check the accuracy of the classifier I used the statistical metrics for Precision, Recall, Accuracy and F1 Score. The Java Collection library classes were used as they made programming a little easier. The Tree was implemented using a class hierarchy of generic node as the parent class of a leaf node and an internal node. The child nodes for an internal node were represented using a collection of nodes.

   (b) [2 points] There was no error when testing on the training data.

   (c) [5 points] There was no error when testing on the testing data.

   (d) [1 points] The maximum tree depth was 3.

2. [15 points] **Limiting Depth**

   (a) [10 points]

| *Depth* | *Average Accuracy* | *Standard Deviation* |
|---|---|---|
| 1 | 0.983 | 0.037 |
| 2 | 0.997 | 0.007 |
| 3 | 1.0 | 0.0 |
| 4 | 1.0 | 0.0 |
| 5 | 1.0 | 0.0 |
| 10 | 1.0 | 0.0 |
| 15 | 1.0 | 0.0 |
| 20 | 1.0 | 0.0 |

(b) [5 points] A maximum depth of 3 was specified for the decision tree since it was the smallest tree that resulted in an average accuracy of 1.0 with 6-fold cross validation. With training and test data and a depth of 3, the accuracy obtained was 1.0.

## Setting B [25 points]

1. [10 points] **Experiments**

   (a) [2 points] The error after training on setting B training data and also testing on the same data was 0.

   (b) [2 points] On testing on setting B test data, the error was 0.062.

   (c) [2 points] Error on testing using setting A training data was 0.418.

   (d) [2 points] Error on testing using setting A test data was 0.002.

   (e) [1 points] Maximum tree depth was 9.

2. [15 points] **Limiting Depth**

   (a) [10 points]

| *Depth* | *Average Accuracy* | *Standard Deviation* |
|---|---|---|
| 1 | 0.935 | 0.029 |
| 2 | 0.950 | 0.008 |
| 3 | 0.965 | 0.014 |
| 4 | 0.972 | 0.027 |
| 5 | 0.982 | 0.024 |
| 10 | 0.988 | 0.026 |
| 15 | 0.988 | 0.026 |
| 20 | 0.988 | 0.026 |

The depth with the greatest accuracy was seen to be 10. At higher depths, the accuracy remained the same. For this reason, the best depth chosen should be 10. The reason for not choosing larger depths which give the same accuracy is that trees with smaller depths will generalize better and thus reduce overfitting.

   (b) [5 points] The depth with the greatest accuracy was 10. An accuracy of 0.938 was obtained.

### Setting C (CS 6350 Students) [20 points]

1. [5 points] To handle methods 1 and 2, the data was pre-processed before being used for training. The '?' character was replaced by most commonly occurring value for the feature in the case of method 1. For method 2, the '?' character was replaced by the most commonly occurring feature value with the same label as the row that had the '?' character. For method 3, the training logic was updated to consider '?' as one of the possible values for the features that happened to contain this character.

2. [10 points] Based on the results, it looks like the feature that had '?' as the value in some places was not part of the tree, since all three methods returned the same result. The tree height reported was 4. That was the reason that depth settings of 4 and above had an accuracy of 1.0.

| Depth | Method 1 | | Method 2 | | Method 3 | |
|---|---|---|---|---|---|---|
| | Avg Acc | Std Dev | Avg Acc | Std Dev | Avg Acc | Std Dev |
| 1 | 0.984 | 0.025 | 0.984 | 0.025 | 0.984 | 0.025 |
| 2 | 0.994 | 0.007 | 0.994 | 0.007 | 0.994 | 0.007 |
| 3 | 0.993 | 0.009 | 0.993 | 0.009 | 0.993 | 0.009 |
| 4 | 1.0 | 0.0 | 1.0 | 0.0 | 1.0 | 0.0 |
| 5 | 1.0 | 0.0 | 1.0 | 0.0 | 1.0 | 0.0 |
| 10 | 1.0 | 0.0 | 1.0 | 0.0 | 1.0 | 0.0 |
| 15 | 1.0 | 0.0 | 1.0 | 0.0 | 1.0 | 0.0 |
| 20 | 1.0 | 0.0 | 1.0 | 0.0 | 1.0 | 0.0 |

3. [5 points] Method 1 was chosen and the accuracy obtained was 1.0.

## References

[1] Mitchell, Tom M. "Decision Tree Learning." *Machine Learning.* New York: McGraw-Hill, 1997. N. pag. Print.