This week we return to classification and explore a non-parametric technique for classifier learning: top-down induction of decision trees.

1 Decision Trees and the C4.5 Algorithm

We will focus our attention on a particular decision tree learning algorithm: C4.5, which builds decision trees by recursively selecting attributes on which to split. The criterion used for selecting an attribute is information gain.

1.1 Reading

There are many good sources of information on decision trees and the C4.5 algorithm. You might want to quickly read Alpaydin, Sections 9.1-9.3 first. Then I recommend Mitchell, Chapter 3, which should be your primary source for this topic. If you’re interested in other sources, you can also look at the following:

- Russell and Norvig, Section 18.3;
- Ross Quinlan’s paper “Induction of Decision Trees”, which appeared in Volume 1 of the journal Machine Learning;
- Ross Quinlan’s book, C4.5: Programs for Machine Learning, which you’ll find with the machine learning books in the lab.

1.2 C4.5

You won’t be doing any implementation of your own this week. But you might find it useful to run C4.5 to get a sense of what it does and how well it works. To do this, you’ll run Weka – open source software for data mining written in Java. (The Witten and Frank book in the lab is a companion to an earlier version of Weka.) Weka is already installed on the machines in the unix lab. To run it, simply type

```
java -Xmx1g -jar /usr/share/java/weka.jar
```

This will start up a GUI. Click on the button that’s labelled “Explorer”. This will open another window that will allow you to select machine learning algorithms and datasets on which you can test them.

Begin by selecting a data set. You can do this by clicking on “Open file...” and then selecting an “arff” file of your choice. I’ve put some interesting data in my shared cs374 directory:

```
/home/faculty/andrea/shared/cs374/
```

You can select any file with an “arff” extension. You’ll see some at the top level of the cs374 directory. Still more can be found in the UCI directory.

Once you’ve selected a data file, click on “Classify” and then “Choose”. In the “Trees” directory, click on J48 (which is really C4.5). Then click on “Start” at the left side of the window below “Test Options”. (I had to resize the window to see the start button.) The output of the classifier will appear in the right half of the window.

If you have any trouble working with Weka, let me know.

1.3 Exercises

1. (From Dietterich) Consider the following decision tree:
(a) Draw the decision boundaries defined by this tree. Each leaf is labeled with a letter. Write this letter in the corresponding region of instance space.

(b) Give another decision tree that is syntactically different but defines the same decision boundaries.

2. (From an exercise by Terran Lane) Consider a two-category classification task with the following training data:

<table>
<thead>
<tr>
<th>attr_1</th>
<th>attr_2</th>
<th>attr_3</th>
<th>attr_4</th>
<th>class</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1</td>
<td>c</td>
<td>-1</td>
<td>c_1</td>
</tr>
<tr>
<td>b</td>
<td>0</td>
<td>c</td>
<td>-1</td>
<td>c_1</td>
</tr>
<tr>
<td>a</td>
<td>0</td>
<td>c</td>
<td>1</td>
<td>c_1</td>
</tr>
<tr>
<td>b</td>
<td>1</td>
<td>c</td>
<td>1</td>
<td>c_1</td>
</tr>
<tr>
<td>b</td>
<td>0</td>
<td>c</td>
<td>1</td>
<td>c_2</td>
</tr>
<tr>
<td>a</td>
<td>0</td>
<td>a</td>
<td>-1</td>
<td>c_2</td>
</tr>
<tr>
<td>a</td>
<td>1</td>
<td>a</td>
<td>-1</td>
<td>c_2</td>
</tr>
<tr>
<td>b</td>
<td>1</td>
<td>c</td>
<td>-1</td>
<td>c_2</td>
</tr>
</tbody>
</table>

Construct a complete (unpruned) decision tree for this data using information gain as your splitting criterion. Please show all entropy calculations.

3. (Modified from Russell and Norvig) In the recursive construction of decision trees, it sometimes happens that a mixed set of positive and negative examples remains at a leaf node, even after all the attributes have been used.

Suppose that you have learned a decision tree for a particular two-class problem, where 1 represents the positive class and 0 represents the negative class. Furthermore, assume that you have p positive examples and n negative examples at the leaf.

(a) Show that the class probability \( p/(p + n) \) minimizes the sum of squared errors.

(b) Show that the solution which picks the majority class minimizes the absolute error over the set of examples at the leaf.

4. This exercise will have you consider an interesting property of the entropy function.

For all parts of this exercise, you should assume a binary classification task, where all attributes are binary as well.

(a) Show that the entropy function is concave. Yes, I know there are pictures of it in both Alpaydin and Mitchell. I still want you to dig out your old calc skills. (If you do this by finding the second derivative of the entropy function, you’ll find it convenient to substitute \( \ln \) for \( \log_2 \).

(b) Suppose that a binary-valued attribute splits a set of examples \( E \) into subsets \( E_1 \) and \( E_2 \), and that the subsets have \( p_1 \) and \( p_2 \) positive examples and \( n_1 \) and \( n_2 \) negative examples, respectively. Show that the attribute has 0 information gain if the ratios \( p_1/(p_1 + n_1) \) and \( p_2/(p_2 + n_2) \) are the same.
A function $f(x)$ is concave on an interval $[a, b]$ if for any two points $x_1$ and $x_2$ in $[a, b]$ and any $\lambda$, where $0 < \lambda < 1$,

$$f(\lambda x_1 + (1 - \lambda)x_2) \geq \lambda f(x_1) + (1 - \lambda)f(x_2)$$

That is, the value at the midpoint of every interval in the domain exceeds the average of its values at the ends of the interval.
Use this to show that every attribute has non-negative information gain.

(d) What does part c imply about the information content of the data and about the process of constructing a decision tree?

## 2 Difficult Data

Of course, no learning algorithm is perfect. Decision tree learning algorithms have difficulty handling certain types of “hard” data.

### 2.1 Reading

Read “Skewing: An Efficient Alternative to Lookahead for Decision Tree Induction” by Page and Ray, which appeared in IJCAI-03.

### 2.2 Exercise

Write a thorough summary and critical analysis of the “Skewing” paper. For the tutorial session you should be prepared to present your summary and critique and to discuss it with your partner and with me.