

CS 441/541
Artificial Intelligence
Fall, 2008

Homework 4:
Decision Trees and Naive Bayes Classification

Due Wednesday, October 29.

In this homework assignment you will compare the classification abilities and robustness of decision trees and naive Bayes classifiers.

Part I: Decision Trees

C4.5 is an implementation of the ID3 Decision Tree Induction algorithm written by Ross Quinlan and available at

<http://www2.cs.uregina.ca/~hamilton/courses/831/notes/ml/dtrees/c4.5/tutorial.html>

(linked from our class web page).

You will be using the spam data linked from our class web page:

<http://web.cecs.pdx.edu/~mm/ArtificialIntelligenceFall2008/spambase.data>
<http://web.cecs.pdx.edu/~mm/ArtificialIntelligenceFall2008/spambase.test>
<http://web.cecs.pdx.edu/~mm/ArtificialIntelligenceFall2008/spambase.names>

For part I of this homework, you need to perform the following steps:

- **Download C4.5:** Download C4.5 and documentation from the site listed above.

This version of C4.5 works on Linux systems. If you have any trouble getting it to compile and work, let me know.

For this assignment, you will need only the manual pages for “c4.5” and for “verbose c4.5”.

- **Download spam data files:** Download “spambase.data”, “spambase.test”, and “spambase.names” from the class web site. In the data and test files, each example is represented by 57 attribute values followed by the class of the example (0 or 1), all on one line.

The training data “spambase.data” contains 1454 positive examples (class 1) and 1454 negative examples (class 0). The test data “spambase.test” contains 359 positive and 359 negative examples.

- **Experiment 1: Run C4.5 on the data:** To run C4.5, type

```
c4.5 -f spambase -u
```

Create a table for your results (of this and following experiments) that lists:

```
attribute at root of tree
% errors on test set (unpruned/pruned)
% false positives (non-spam classified as spam) on test set
% false negatives (spam classified as non-spam) on test set
```

- **Experiment 2: Smaller training set:** Create a new training set, “small-spambase.data” that contains half the positive and negative examples of the original training data. (You need to copy “spambase.test” to “small-spambase.test” and “spambase.names” to “small-spambase.names”.)

Run

```
c4.5 -f small-spambase -u
```

and record the results in your table, as above.

- **Experiment 3: Noisy training set:** Create a new training set, “noisy-spambase.data” into which you introduce some noise. To do this, copy the training examples from “spambase.data” and change approximately 5% of the positive examples’ class from 1 to 0, and approximately 5% of the negative examples’ class from 0 to 1. (Again, copy “spambase.test” to “noisy-spambase.test” and “spambase.names” to “noisy-spambase.names”.)

Run

and record the results in your table as above.

- In your writeup, give your table and write a paragraph summarizing the results of the experiments. Were the results what you expected?

Part II: Naive Bayes Classifier

For this part you will implement a naive Bayes classifier, and compare its performance on classifying spam and non-spam with that of the decision trees you used in Part I.

Here are the steps you need to perform:

- **Discretize the data.** You need to transform the continuous-valued attributes in the `spambase.data` and `spambase.test` files into a small number of discrete bins for each attribute. For each attribute a_i , take the interval defined by $[a_i^{min}, a_i^{max}]$ and divide it into 10 approximately equal-sized bins. (Recall that there are 57 attributes, listed in `spambase.names`.)
- **Train a naive Bayes classifier.** First calculate (from your training data) the prior probabilities of the two classes, $P(1)$ and $P(0)$. Then, for each attribute a_i and bin b_j , calculate $P(a_i \in b_j|0)$ and $P(a_i \in b_j|1)$, over your training set.
- **Experiment 1:** Run your naive Bayes classifier on the test set, `spambase.test`. For each example in the test set, figure out which bin corresponds to each attribute's value. (Here I call this bin $bin(a_i)$.)

Then compute

$$P(0) \prod_{i=1}^{57} P(a_i \in bin(a_i)|0),$$

and

$$P(1) \prod_{i=1}^{57} P(a_i \in bin(a_i)|1).$$

(If any of the $P(a_i \in \text{bin}(a_i)|\text{class}) = 0$, simply replace it with the small value .0014, approximating the “m-estimate of probability” to avoid multiplying by zero.) The classification for that test example is the classification corresponding to the higher of these two numbers.

As before, record

% errors on test set (unpruned/pruned)

% false positives (non-spam classified as spam) on test set

% false negatives (spam classified as non-spam) on test set

- **Experiment 2:** Repeat Experiment 1, but using your small training set.
- **Experiment 3:** Repeat Experiment 2, using your noisy training set.
- In your writeup, give your table with the results of these three experiments and write a paragraph summarizing the results of the experiments. Were the results what you expected? How did the performance of your naive Bayes classifier compare with that of your decision tree classifier?

Grading

You will be graded on the completeness and clarity with which you report and explain your results on the steps listed above.