# CS 440: Introduction to AI

Homework 5

Due: November 30 11:59PM, 2010

*Your answers must be concise and clear. Explain sufficiently that we can easily determine what you understand. We will give more points for a brief interesting discussion with no answer than for a bluffing answer.*

*Please email your solution to the TA at cs440ta@cs.illinois.edu.*

## 1 Naive Bayes

| MPG | Horsepower | Weight | Acceleration | Displacement |
|---|---|---|---|---|
| High | LT100 | Light | Wow | Small |
| Low | GT100 | Heavy | Poor | Large |
| Moderate | LT100 | Light | Wimpy | Medium |
| Moderate | LT100 | Light | Wimpy | Medium |
| High | LT100 | Light | Wimpy | Small |
| Low | GT100 | Heavy | Wimpy | Medium |
| Moderate | LT100 | Heavy | Wimpy | Medium |
| Low | GT100 | Heavy | Wimpy | Large |
| Moderate | LT100 | Heavy | Poor | Medium |
| Moderate | GT100 | Heavy | Wow | Large |
| Low | GT100 | Heavy | Poor | Large |
| Moderate | GT100 | Light | Wimpy | Medium |
| Low | GT100 | Heavy | Wimpy | Large |
| Low | GT100 | Heavy | Wow | Medium |
| High | LT100 | Light | Wimpy | Small |
| Moderate | GT100 | Heavy | Poor | Medium |

Table 1: A data set. LT100: Less than 100 HP. GT100: Greater than 100 HP.

Table 1 shows an automobile dataset. The data consists of five features (MPG, Horsepower, Weight, Acceleration, Displacement). Each is binary or trinary. Assume for your Nave Bayes net that *MPG, Horsepower, Weight, Acceleration* are conditionally independent from each other given *Displacement*. We will learn a Nave Bayes net from the dataset in Table 1 and use it to estimate missing values of new examples

In naive Bayes net, attributes ($X$) are conditionally independent of each other given a class variable ($Y$). $Y$ can be infered by the following Naive Bayes classification rule:

$$Y \leftarrow \underset{y_k}{\operatorname{argmax}} P(Y = y_k) \prod_i P(X_i \mid Y = y_k) \tag{1}$$

In order to estimate parameters, we use Maximum Likelihood Estimation (MLE) with *add-one smoothing* as follows:

$$\hat{P}(Y = y_k) = \frac{\#D\{Y = y_k\} + 1}{\mid D \mid + J} \tag{2}$$

$$\hat{P}(X_i = x_{ij} \mid Y = y_k) = \frac{\#D\{X_i = x_{ij} \wedge Y = y_k\} + 1}{\#D\{Y = y_k\} + K} \tag{3}$$

where the $\#D\{x\}$ represents the number of elements in the data set D that satisfy property x, $J$ is the number of distinct values $Y$ can take on, and $K$ is the number of distinct values $X_i$ can take on. In addition to our text, you may find it useful to visit `http://www.cs.cmu.edu/~tom/mlbook/NBayesLogReg.pdf`.

1. From all the examples in the table 1, estimate parameters.
   (a) What is the estimated probability (2) for each $y_k$?
   (b) What is the estimated probability (3) for the attribute *Weight* and each $y_k$? (The number of parameters you need to estimate is 6.)
   (c) Explain in your own words why it is useful to add one to the numerator counts in equation (2) and (3).

2. We can classify new examples by Naive Bayes classification (1) with the estimated probabilities. We have a new example ($MPG$:High, $Horsepower$:LT100, $Weight$:Light, *Acceleration*:Poor). To make the problem simple, we provide a fake conditional probability table of $P(X_{ij} \mid Y_k)$ for the values in the example.

| Displacement | MPG:High | Horsepower:LT100 | Weight:Light | Acceleration:Poor |
|---|---|---|---|---|
| Small | 0.5 | 0.6 | 0.5 | 0.1 |
| Medium | 0.2 | 0.5 | 0.4 | 0.3 |
| Large | 0.1 | 0.3 | 0.2 | 0.4 |

Table 2: Fake conditional probability table

Classify the new example by Naive Bayes classification (1) with the probabilities obtained from 1-(a) and the fake probabilities in table 2. Show your work.

## 2 Decision Tree

1. For this question, you will manually induce a decision tree from the training data, which are the first twelve examples in table 1. Then, we will predict *MPG* with the other four attributes in the test data (remaining four examples). *Use the order of the attributes in the table to break the tie.* You can report the decision tree as a series of `if-then` statements, where each indentation is a node level in the tree, as follows:

```
if attribute 1 = x
    if attribute 2 = y
        if attribute 3 = z
            class = +
        if attribute 3 = q
            class = 0
        if attribute 3 = r
            class = -
    if attribute 2 = s
        class = +
if attribute 1 = t
    if attribute 3 = z
        class = 0
    if attribute 3 = r
        class = -
```

   (a) What is the entropy of the training data (first twelve examples) over the class variable (*MPG*)? Show your work.

   (b) Use the training data to generate a decision tree. Express your decision tree using a series of `if-then` statements.

   (c) How well do the tree perform on the test data (remaining four examples)? Report the classification results and the error rate (that is, the ratio of the errors to the number of examples).

# 3 Evaluation

1. Suppose you are running a learning experiment on a new algorithm for Boolean classification. You have a data set consisting of 15 positive and 15 negative examples. You plan to use leave-one-out cross validation and compare your algorithm to a baseline function, a simple majority classifier. (A majority classifier simply outputs the most common class label from the dataset without considering any of the examples attribute values.) What is the accuracy of the majority classifier? Briefly justify your answer.