Mark Klobukov

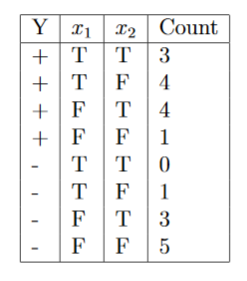
CS 383: Homework 4

Professor Matthew Burlick

2/19/2018

**Part I: Theory**

1) Training examples for an unknown target function:



The dataset has two classes: + and – . The total number of data with class + is (3+4+4+1) = 12.   
Total number of data with class – is (0+1+3+5) = 9

(a) What is the sample entropy from this training data?



b) What are the IGs for branching on variables and ?

(c) Draw the decision tree that would be learned by the ID3 algorithm without pruning from this training data.

Feature has a higher information gain, so place it at the root of the tree. Then split each branch further based on feature . After that, no more splits are possible. Perfect split is not achieved, but we can take the mode class in a given leaf and make that the final decision.

F

T

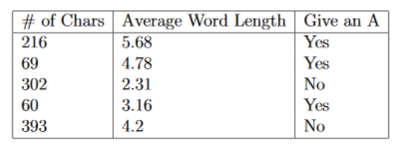
T

F

F

T

2) Five data samples:



(a) What are the class priors?

Class priors are the probabilities of encountering the class in a dataset.

(b) Find parameters of the Gaussians.

Standardize data so that there is no unfair bias toward features of different scales. First column of the following matrix represents standardized number of characters, second column is the average word length (also standardized). In the third column, 1/0 correspond to Yes/No.

Model for “Yes” observations:

# characters:

Avg. word length:

Model for “No” observations:

#characters:

Avg. word length:

(c) Determine if an essay with 242 characters and average word length of 4.56 should get an A.

Standardize 242 characters (208 is mean, 145.2154 is std)

Standardize 4.56 word length: (4.026 is mean, 1.3256 is std)

Compute

By naïve independence assumption (and discarding the denominator):

Let *p* stand for the approximation of P with Gaussian probability density function. normpdf() is a MATLAB-syntax Gaussian PDF function with three arguments: value x, mean, std.

This is not a probability. That is, the number above cannot be interpreted as 5.7 % chance of getting an A. It is necessary to perform the same computation for a grade below an A and then compare the two numbers.

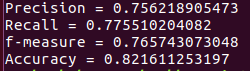
By naïve independence assumption (and discarding the denominator):

The value of likelihood obtained for class A = 0.0574; for class <A = 0.0231

The value for class A is higher, therefore it is about twice as likely that the essay will get an A than that the essay will get below an A. Answer: yes, by our classifier, the essay should get an A.

**Part II: KNNs**

Running the code on the provided data with k = 5 produces the following statistics:



|  |  |  |
| --- | --- | --- |
| **Statistic** | **Value Observed** | **Value from HW Instructions** |
| Precision | 75.622 % |  |
| Recall | 77.551 % |  |
| F-Measure | 76.574 % |  |
| Accuracy | 82.161 % |  |

Table 1: KNNs statistics for the spam data, k = 5

These values are lower than those provided in the homework instructions by 10-15%. It is possible that the randomization happened differently. I provide the zero seed through the **np.random.seed(0)** command. It is possible that a different method was used by the instructor in the algorithm. There also was not a perfect match between numbers in previous assignments.