

CS 374 Assignment #9 Computational Learning Theory

Due the week of April 14, 2008

This week you will revisit a topic that you considered briefly during the first week of the semester – the PAC (Probably Approximately Correct) framework for analyzing learning algorithms. You will look in more detail at issues like finding the number of examples needed to learn a concept. You will also see how the VC dimension fits into this framework. Finally, you will consider a variety of concept classes and see how they can be proven to be PAC-Learnable.

A number of important machine learning algorithms have their roots in the PAC model and, more generally, in the computational learning theory community. Among these is AdaBoost, one implementation of the general framework of Boosting. This week you will read about and implement AdaBoost.

1 Learning Theory

1.1 Reading

Please read the following:

- Mitchell, Chapter 7 (Computational Learning Theory), up to page 220,
- Kearns and Vazirani, pages 22-24 (Using 3-CNF Formulae to Avoid Intractability). While it isn't absolutely necessary, you might find it even better to read pages 16-24.

1.2 Exercises

Please do the following exercises:

- From Mitchell, 7.1, 7.3, 7.5 b, 7.7 a & b.
- At the bottom of page 24 of Kearns and Vazirani, it says that the mapping from one feature space to another distorts the distribution from which instances are drawn for learning. As an example, it says that if D is a uniform distribution over $\{0,1\}^n$, D' is not necessarily uniform over the transformed assignments. Demonstrate this.

2 AdaBoost

As described above, the AdaBoost algorithm is one of several that have grown out of the PAC learning model. This week you will learning about this algorithm through reading and an implementation exercise. (Next week you will consider Boosting more generally, as well as other methods that involve ensembles of classifiers.)

2.1 Reading

Please read “A Short Introduction to Boosting” by Yoav Freund and Robert E. Schapire.

I expect that you will find this paper to be quite clear, though there are some items on page 6 that you might find a little bit confusing. The basic idea described there is as follows. Since AdaBoost

concentrates on training examples that are close to the margin, it naturally increases the margin over time.

Also, note that “OCR” stands for “Optical Character Recognition”. The authors assume that this is obvious to the reader, which is reasonable within the machine learning community, but some of you might not yet be familiar with this.

2.2 Implementation

Please implement the AdaBoost algorithm as given on page 3 of the Freund and Schapire paper. The algorithm requires that you train a weak learner on data sampled from the training set. While I expect you to design your AdaBoost program in such a way that you can plug in any weak learner, I would like you to use Decision Stumps this week. Decision Stumps are simply one-level decision trees. That is, the learner selects an attribute for the root of the tree and immediately classifies examples based on their values for that attribute. While you will need to modify your decision tree programs a bit to do this, you obviously have most of the code already.

In order to test your program, you can use the `weather.nominal.arff` and `contact-lenses.arff` files, as neither is missing any values. Of course, you should feel free to implement a decision stump learner that can handle missing values. Because these data sets are so small, perform a leave-one-out evaluation.

As usual, please turn in paper copy as well as electronic copy.