15-850(C) Machine Learning Theory

Assignment 3 Due: October 24, 1995

Variations on the Winnow algorithm

1. **A bad modification.** Suppose that we modify Winnow so that it doubles its weights on positive examples even when it did *not* make a mistake. Show how this can cause the algorithm to make an unbounded number of mistakes.

2. **Tracking a moving target.** Imagine playing the following game against an adversary. We have n variables and begin with the target concept that is the disjunction of zero of them (it says everything is negative). Each round of the game goes like this:

   Adversary's turn: The adversary may change the target concept by adding or removing some variables from the target disjunction. The adversary has to pay \$1 for each variable added, though removing variables is free. It then presents an example to the learning algorithm.

   Learner's turn: The learning algorithm makes a prediction on the example given, and then is told the correct answer (according to the current target concept) and is charged \$1 if it made a mistake.

   Consider the variation on the Winnow algorithm that never lets any weight get below 1/2 (when a mistake is made on a negative example, only weights of value 1 or more are cut in half). Let A denote the adversary's cost so far, and let L denote the learner's cost so far. Prove that this Winnow variant guarantees $L = O(A \log n)$. So, you can think of the result proven in class as a special case of this.

   Hint. Think of doubling a (currently) relevant weight as placing a poker chip on that weight. If r is the number of currently relevant variables, consider the "potential function":

   $$r \log(2n) - (\text{\# chips on relevant variables})$$

   which can never go below zero.

3. **Combining sources of advice.** Consider the following scenario. We have an algorithm which has access to n "advisors" (think of an advisor as a variable or feature). On each example, some advisors make a prediction and some abstain. Based on these responses, the learning algorithm must make a prediction, after which it is told the correct answer, charged a penalty of 1 if it made a mistake, and then the process repeats. Let's suppose that out of the n advisors, there is some subset of r "good" ones such that (A) none of the good ones ever produces an incorrect prediction, and (B) on every example at least one good advisor predicts something.

   We will use the following Winnow variant. Initialize all advisors to start with weight 1. Given an example, poll the advisors and produce the prediction that has the highest weighted vote. If a mistake is made, multiply the weights of advisors that predicted incorrectly by 1/2, multiply the weights of advisors that predicted correctly by 3/2, and leave those that abstained alone.

   Prove that the maximum number of mistakes made by this algorithm is $r \log_{3/2}(n/r)$.

   Hint. Use the fact that the total weight of good advisors cannot be larger than the total weight of all the advisors.