15-859(A) Machine Learning Theory

Homework # 5 **Due: April 1, 2004**

Exercises:

1. A *distinguishing sequence* for a DFA is a sequence of actions such that the observations produced from these actions uniquely determines the starting state. I.e., a sequence h such that if $q \neq q'$ then $\text{obs}(q, h) \neq \text{obs}(q', h)$.

   (a) Describe a DFA that has no distinguishing sequence. Note that the definition of "$q \neq q'$" is that there must exist a sequence $h_{qq'}$ such that $\text{obs}(q, h_{qq'}) \neq \text{obs}(q', h_{qq'})$, it's just that no single h works for all pairs.

   (b) Give a homing sequence for your DFA.

Problems:

In the rest of this homework you will show that the class of polynomial-size boolean formulas is equivalent to the class NC^1. NC^1 is the class of $O(\log n)$-depth {AND, OR, NOT} circuits where each gate has at most 2 inputs. Boolean formulas are just the generalization of DNF in which we allow ANDs and ORs to appear in any order, e.g., $x_1 \vee (x_2 \wedge (\bar{x}_1 \vee x_3))$; you can think of a Boolean formula as a circuit that looks like a tree, except the inputs are allowed to have out-degree greater than 1. This is also exercise 6.2 (p.141) in the book.

2. Show that for any circuit of depth d of {AND, OR, NOT} gates with fanin ≤ 2, there is an equivalent boolean formula of size $O(2^d)$. Thus, NC^1 is contained in the class of polynomial-size boolean formulas.

3. Show that for any boolean formula of size s, there exists an equivalent circuit of depth $O(\log s)$. Thus polynomial-size boolean formulas are contained in the class NC^1.

Thus, this implies our hardness results for learning NC^1 carry over to general Boolean formulas (can't even weak-learn in poly-time over the uniform distribution, even with membership queries).

Note: question 2 should be pretty straightforward. Question 3 is trickier.