0368.4163 Randomness and Computation	May 20, 2009

Homework 4

Lecturer: Ronitt Rubinfeld	*Due Date: June 10, 2009*

Homework guidelines: Same as for homework 1.

1. *Dictator functions*, also called *projection functions*, are the functions mapping $\{+1,-1\}^n$ to $\{+1,-1\}$ of the form $f(x) = x_i$ for i in $[n]$.

   Consider the following test for whether a function f is a dictator: Given parameter δ, the test chooses $x, y, z \in \{1,-1\}^n$ by first choosing x, y uniformly from $\{1,-1\}^n$, next choosing w by setting each bit w_i to -1 with probability δ and $+1$ with probability $1-\delta$ (independently for each i), and finally setting z to be $x \circ y \circ w$, where $\circ$ denotes the bitwise multiply operation. Finally, the test accepts if $f(x)f(y)f(z) = 1$ and rejects otherwise.

   - Show that the probability that the test accepts is $\frac{1}{2} + \frac{1}{2}\sum_{s\subseteq[n]}(1-2\delta)^{|S|}\hat{f}(S)^3$.
   - Show that if f is a dictator function, then f passes with probability at least $1-\delta$.
   - Show that if f passes with probability at least $1-\epsilon$ then there is some S such that $\hat{f}(S)$ is at least $1-2\epsilon$ and such that f is ϵ-close to χ_S.
   - Why isn't this enough to give a dictator test? (i.e., what nondictators might pass?) Give a simple fix.

2. Show that if there is a PAC learning algorithm for a class C (sample complexity $poly(\log n, 1/\epsilon, 1/\delta)$) then there is a PAC learning algorithm for C with sample complexity dependence on δ (the confidence parameter) that is only $\log 1/\delta$ – i.e., the "new" PAC algorithm should have sample complexity $poly(\log n, 1/\epsilon, \log 1/\delta)$. (It is ok to assume that the learning algorithm is over the uniform distribution on inputs, although the claim is true in general.)

3. - Show that for any monotone function $f : \{+1,-1\}^n \to \{+1,-1\}$, the influence of the i^{th} variable is equal to the value of the Fourier coefficient of $\{i\}$, that is $\text{inf}_i(f) = \hat{f}(\{i\})$.
   - Show that the majority function $f(x) = \text{sign}(\sum_i x_i)$ maximizes the total influence among n-variable monotone functions mapping $\{+1,-1\}^n$ to $\{+1,-1\}$, for n odd.

4. Consider the sample complexity required to learn the class of monotone functions mapping $\{+1,-1\}^n$ to $\{+1,-1\}$ over the uniform distribution (without queries).

   (a) Show that
   $$\sum_{|S|\geq \text{Inf}(f)/\epsilon} \hat{f}(S)^2 \leq C \cdot \epsilon$$
   where $\text{Inf}(f)$ is the influence of f, and C is an absolute constant.

   (b) Show that the class of monotone functions can be learned to accuracy ϵ with $n^{\Theta(\sqrt{n}/\epsilon)} = 2^{\tilde{O}(\sqrt{n}/\epsilon)}$ samples under the uniform distribution.

Useful definitions:

1. For $x = (x_1, \ldots, x_n) \in \{+1, -1\}^n$, $x^{\oplus i}$ is x with the i-th bit flipped, that is,

$$x^{\oplus i} = (x_1, \ldots, x_{i-1}, -x_i, x_{i+1}, \ldots, x_n).$$

The *influence of the i-th variable on* $f : \{+1, -1\}^n \to \{+1, -1\}$ is

$$\mathrm{Inf}_i(f) = \Pr_x \left[f(x) \neq f\left(x^{\oplus i}\right) \right].$$

The *total influence* of f is

$$\mathrm{Inf}(f) = \sum_{i=1}^{n} \mathrm{Inf}_i(f).$$

2. A function $f : \{+1, -1\}^n \to \{+1, -1\}$ is *monotone* if for all $x, y \in \{+1, -1\}^n$ such that $x_i \leq y_i$ for each i, $f(x) \leq f(y)$. Assume that $-1 < +1$.