See the course Web page for instructions on how to submit homework. **Important:** To make life easier for the TAs, please start each problem on a new sheet of paper.

**Problem 1** Let $C$ be a class of efficiently evaluable hypothesis representations over $\{0,1\}^n$ (a concrete example of such a class $C$ would be the class of all 100-term DNFs, or the class of all linear threshold functions where all of the weights $w_1,\ldots,w_n$ belong to $\{0,1\}$). We define the $C$-Consistency problem as follows:

**Input:** A collection of labeled examples $(x_1,y_1),\ldots,(x_m,y_m)$ where each $x_i \in \{0,1\}^n$ and each $y_i \in \{0,1\}$.

**Output:** “yes” if there is some $c \in C$ such that $y_i = c(x_i)$ for all $i \in \{1,\ldots,m\}$, “no” otherwise.

Suppose that there is no poly$(m,n)$-time randomized algorithm for the $C$-Consistency problem. Show that then there is no efficient proper PAC learning algorithm for $C$.

**Problem 2** Given any $w \in \{0,1\}^n$ and any integer $\theta \geq 0$, the 0/1-weight halfspace $TH_{\theta,w} : \{0,1\}^n \to \{0,1\}$ is defined as follows: given input $x \in \{0,1\}^n$, we have $TH_{\theta,w}(x) = 1$ iff $w \cdot x \geq \theta$, where $w \cdot x$ denotes the standard real-valued dot product of two $n$-dimensional vectors. So a 0/1-weight halfspace is simply a linear threshold function in which the weights are required to be 0 or 1 and the threshold is an integer — in fact, it is simply a “$\theta$-out-of-$|w|$” function, where $|w|$ denotes the number of 1s in the vector $w$.

The Binary Programming problem is defined as follows: An instance is a set of $s$ pairs $\langle c_i,b_i \rangle$ and the pair $\langle \pi,B \rangle$, where $c_i \in \{0,1\}^n, \pi \in \{0,1\}^n, b_i \in \{0,1\}$, and $0 \leq B \leq n$. The problem is to determine whether there exists a vector $\vec{d} \in \{0,1\}^n$ such that $\pi \cdot \vec{d} \leq b_i$ for $1 \leq i \leq s$ and $\pi \cdot \vec{d} \geq B$.

Prove that if there is a polynomial time proper PAC learning algorithm for the class of 0/1-weight halfspaces, then there is a polynomial time randomized algorithm which solves the Binary Programming problem. (The Binary Programming problem is known to be NP-complete, so it’s very unlikely that there is an efficient randomized algorithm that can solve it.)

**Problem 3** In this problem we consider the instance space $X = \{1,2,\ldots,N\}$. An arithmetic progression over $X$ is a subset of $X$ of the form $X \cap \{a + bi : i = 0,1,2,\ldots\}$ where $a$ and $b$ are natural numbers. For instance, for $N = 30$ the set $\{7,12,17,22,27\}$ is an arithmetic progression over $\{1,\ldots,N\}$.

Let $C$ be the concept class consisting of all arithmetic progressions over $\{1,\ldots,N\}$.
(i) Show that the VC dimension of $C$ is $O(\log N)$.

(ii) Show that the VC dimension of $C$ is $\Omega(\log N/\log \log N)$. (Hint: The fact that the product of all prime numbers up to $k$ is $2^{\Theta(k)}$ may be useful, and also the Prime Number Theorem.)

**Problem 4** (You may assume the results of problem 3 for this problem even if you did not solve problem 3.)

Taking $N \to \infty$, part (ii) of problem 3 implies that there is no a priori fixed sample size which suffices for PAC learning the concept class $C$ of all arithmetic progressions over the infinite domain $\mathbb{N}$ for all distributions. To be more precise, it tells us that there is no function $m(1/\varepsilon, 1/\delta)$ such that the following holds: There is an algorithm which, given $\varepsilon, \delta$ and access to $EX(c, D)$ where $D$ is any distribution over $\mathbb{N}$ and $c$ is any arithmetic progression over $\mathbb{N}$, draws $m(1/\varepsilon, 1/\delta)$ samples from $EX(c, D)$ and with probability $1 - \delta$ outputs an $\varepsilon$-accurate hypothesis for $c$.

However, while there is no fixed sample size $m(1/\varepsilon, 1/\delta)$ that suffices for every distribution, in fact for every distribution there is some sample size that suffices. Establishing this is the point of the current problem.

Show that the following holds: For every distribution $D$ over $\mathbb{N}$, there is a function $m_D(1/\varepsilon, 1/\delta)$ (which may depend on $D$) and an algorithm $A_D$ (which also may depend on $D$) such that the following holds: If $A_D$ is given $\varepsilon, \delta$ and access to $EX(c, D)$ where $c$ is any arithmetic progression over $\mathbb{N}$, it draws $m_D(1/\varepsilon, 1/\delta)$ samples from $EX(c, D)$ and with probability $1 - \delta$ outputs an $\varepsilon$-accurate hypothesis for $c$.

**Problem 5** Recall that a halfspace is a Boolean function of the form $f(x) = \text{sign}(w_1 x_1 + \cdots + w_n x_n - \theta)$, where $\text{sign}(z) = 1$ if $z \geq 0$ and $\text{sign}(z) = -1$ if $z < 0$. Let $C$ be the class of all functions over $\{0, 1\}^n$ which are unions of $k$ halfspaces, i.e.

$$C = \{ f : \exists \text{ halfspaces } h_1, \ldots, h_k \text{ such that } f(x) = h_1(x) \lor \cdots \lor h_k(x) \text{ for all } x \in \{0, 1\}^n \}.$$ 

Show that the VC dimension of $C$ is at most $O(nk \log k)$. (Hint: Think about the growth function $\Pi_C(m)$. You may use the fact, stated in class, that the VC dimension of the class of $n$-variable halfspaces is $n + 1$.)