Leslie G. Valiant
TFs: Aayush Karan and Kevin Cong

CS 2280: Computational Learning Theory
Homework 3 Solutions
Due: Mar. 28, 11:59pm

Policy reminders You are strongly encouraged to type your solutions using LaTeX. You may discuss problems with your classmates, but not merely copy each others solutions. You must write all solutions by yourself, list your collaborators on your problem sets and also appropriately cite any resources outside of the class materials that you have used. You are not allowed to look up solutions to the problems. Please do not use LLMs or LLM-assisted tools for finding solutions to the problems.

Problem 1. (10pt) **Saturating Sauer-Shelah.** Show that for any d there is a concept class $\mathcal{C}$ of VC dimension d such that for any m there exists a set S of m points such that $|\Pi_{\mathcal{C}}(S)| = \Phi_d(m)$.

Problem 2. (10pt) **Monotone Boolean functions are not PAC-learnable.** Let $x = x_1 \dots x_n \in \{0,1\}^n$ and $y = y_1 \dots y_n \in \{0,1\}^n$ be two n-bit strings. We say that $x \geq y$ if $x_i \geq y_i$ for all i. A boolean function $f : \{0,1\}^n \to \{0,1\}$ on variables $x_1, \dots, x_n$ is said to be *monotone* if $x \geq y$ implies $f(x) \geq f(y)$.
Let M_n denote the class of all monotone Boolean functions over $\{0,1\}^n$. Prove that there is no PAC learning algorithm for M_n whose running time is a polynomial function of $n, \frac{1}{\epsilon}, \frac{1}{\delta}$ (without size(c)).

Problem 3. (10pt) **VC-dimension of parity functions.** Define the class of parity functions $\mathcal{P}$ over $X = \{0,1\}^n$ as follows: Let $a \in \{0,1\}^n$, then $\chi_a(x) = 1$ if $a \cdot x$ is odd and $\chi_a(x) = 0$ otherwise, where $a \cdot x = \sum_{i=1}^n a_i x_i$. Prove that the VC-dimension of $\mathcal{P}$ is n.

Problem 4. (10pt) **Compositional VC Dimension.** Let $\mathcal{C}$ be a concept class over some domain X and $\mathcal{F}_T$ be a concept class over $\{0,1\}^T$. We define a class of functions $\mathcal{F}_T(\mathcal{C})$ over X as follows.

$$\mathcal{F}_T(\mathcal{C}) = \{g(c_1(x), c_2(x), ..., c_T(x)) | g \in \mathcal{F}_T \;\text{ and }\; c_1, ..., c_T \in C\}.$$

Prove that VC-dim$(\mathcal{F}_T(\mathcal{C})) = O(\ell \log \ell)$ where $\ell =$ VC-dim$(\mathcal{F}_T) + T\cdot$VC-dim$(\mathcal{C})$.

Problem 5. (15pt) **Occam as a weak learning algorithm.** Let $\mathcal{C}$ be any concept class. Show that if there exists an (α, β)-Occam algorithm for $\mathcal{C}$, then there exists an efficient randomized Occam algorithm that given sample S of size m for $c \in C$, with probability at least $1 - \delta$, outputs a hypothesis h consistent with S such that size$(h) \leq p(n, size(c), \log m)$ for some polynomial p.

Hint. You can assume here that the description of each real number that occurs in an execution of the Adaboost algorithm takes $O(1)$ space.

Problem 6. (15pt) **Adaboost on weak learning algorithm.** Let $\mathcal{C}$ be a concept class and `WeakLearn` be an algorithm that weakly PAC learns $\mathcal{C}$ and generates hypotheses that have error of at most $1/2 - \gamma$ for some positive γ (assume for simplicity that `WeakLearn` always succeeds). Let x be any point in the sample S of size $N \geq 2$.

1. Show that in the Adaboost algorithm, the error of hypothesis h_t on distribution D_{t+1} is exactly 1/2.

2. What is the maximum probability that the Adaboost algorithm can assign to point x in any of the boosting stages?

3. Assuming that `WeakLearn` fails, for as long as it possibly can, to return the correct label for x, what is the maximum number of stages that it will take the Adaboost algorithm to force `WeakLearn` to return a hypothesis which is correct on x (give the best upper bound you can). *You can assume that initially every point has the same probability.*