COMS 4995-1 S20 Homework 1 (due February 19, 2020)

Instructions

Submit your write-up on Gradescope as a neatly typeset PDF document by 11:00 PM of the due date. Please use TEX, LATEX, or a similar system.

On Gradescope, be sure to select the pages containing your answer for each problem. More details can be found on the Gradescope Student Workflow help page.

(If you don't select pages containing your answer to a problem, you'll receive a zero for that problem.)

Also, please make sure that your "Student ID #" on Gradescope is set to your UNI, using only lowercase letters and numbers (e.g., abc1234).

Finally, please make sure **your name and your UNI** appear prominently on the first page of your write-up.

Problem 1 (20 points)

Let X be a non-negative random variable, and let $t > 0$. Recall that Markov's inequality can be directly applied to the k-th power of X to obtain a tail inequality:

$$\Pr(X \geq t) \leq \frac{\mathbb{E}[X^k]}{t^k}.$$

The "Chernoff method" obtains a tail inequality by using all moments of X simultaneously through an exponential: for $\lambda > 0$,

$$\Pr(X \geq t) \leq \frac{\mathbb{E}[e^{\lambda X}]}{e^{\lambda t}}.$$

Prove that

$$\inf_{k \in \mathbb{N}} \frac{\mathbb{E}[X^k]}{t^k} \leq \inf_{\lambda > 0} \frac{\mathbb{E}[e^{\lambda X}]}{e^{\lambda t}}.$$

(This means that the Chernoff method cannot give a smaller tail bound than what can be obtained by optimally choosing the moment order to use with Markov's inequality. One should regard the Chernoff method "just" as a trick to simplify analysis because exponential functions are "nice".)

For simplicity, you may assume that the moment generating function of X exists for all λ.

Problem 2 (20 points)

Another important concentration of measure inequality for independent random variables is *McDiarmid's inequality.*

Theorem (McDiarmid's inequality). Let $X_1, \ldots, X_n$ be independent random variables, where X_i has range $\mathcal{X}_i$. Let $f\colon \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \to \mathbb{R}$ be any function with the $(c_1, \ldots, c_n)$-*bounded differences property*: for every $i = 1, \ldots, n$ and every $(x_1, \ldots, x_n), (x'_1, \ldots, x'_n) \in \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ that differ only in the i-th coordinate ($x_j = x'_j$ for all $j \neq i$), we have

$$|f(x_1, \ldots, x_n) - f(x'_1, \ldots, x'_n)| \leq c_i.$$

For any $t > 0$,

$$\Pr(f(X_1, \ldots, X_n) - \mathbb{E}[f(X_1, \ldots, X_n)] \geq t) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n c_i^2}\right).$$

For the proof, please read the handout on the course website. Check for yourself that Hoeffding's inequality is a simple corollary of McDiarmid's inequality.

(a) Let F be a (possibly infinite) collection of real-valued functions on $\mathcal{X}$, each with range $[a, b]$. Let P be a probability distribution over $\mathcal{X}$, and let P_n be the empirical probability distribution over $\mathcal{X}$ based on an iid sample from P of size n. Use McDiarmid's inequality to prove, for any $t > 0$,

$$\Pr\left(\sup_{f \in F} |Pf - P_n f| - \mathbb{E}\left[\sup_{f \in F} |Pf - P_n f|\right] \geq t\right) \leq \exp\left(-\frac{2nt^2}{(b-a)^2}\right).$$

Above, the notation "Qf" for a probability distribution Q on $\mathcal{X}$ and a real-valued function $f\colon \mathcal{X} \to \mathbb{R}$ is used to denote the expectation of $f(X)$ for $X \sim Q$.

(b) Let $\boldsymbol{p} = (p_1, \ldots, p_k)$ be a probability distribution over $\{1, \ldots, k\}$, and let $\hat{\boldsymbol{p}} = (\hat{p}_1, \ldots, \hat{p}_k)$ be the empirical probability distribution based on an iid sample from $\boldsymbol{p}$ of size n. Prove that, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$,

$$\|\boldsymbol{p} - \hat{\boldsymbol{p}}\|_2 \leq \sqrt{\frac{1 - \|\boldsymbol{p}\|_2^2}{n}} + \sqrt{\frac{\ln(1/\delta)}{n}}.$$

(Note: This can be used to obtain a bound on the total variation distance between $\boldsymbol{p}$ and $\hat{\boldsymbol{p}}$ by bounding l^1-norms by l^2-norms via Cauchy-Schwarz.)

Problem 3 (20 points)

Let $X_1, \ldots, X_n$ be 1-subgaussian random variables (not necessarily independent), and let

$$Z := \max_{i=1,\ldots,n} |X_i|.$$

In this problem, you will prove a bound on $\mathbb{E}[Z]$ two (or three) ways.

(Throughout, assume $n > 1$ so $\log(n) > 0$.)

(a) Prove that, for any $\lambda > 0$,

$$\mathbb{E}[Z] \leq \frac{1}{\lambda} K_Z(\lambda),$$

where $K_Z(\lambda) = \ln \mathbb{E}[\exp(\lambda Z)]$ is the log moment generating function for Z.

(b) Use the result from Part (a) to prove that, for some absolute constant $C > 0$,

$$\mathbb{E}[Z] \leq C\sqrt{\log(n)}.$$

Hint: If $a_1, \ldots, a_n$ are non-negative, then $\max_i a_i \leq \sum_i a_i$.

(c) Prove that for any $t > 0$,

$$\Pr(Z \geq t) \leq 2n \cdot e^{-t^2/2}.$$

Hint: This one is easy; not a trick question.

(d) Use the result from Part (c) and the fact that $\mathbb{E}[Z] = \int_0^\infty \Pr(Z \geq t)\,\mathrm{d}t$ to prove that, for some absolute constant $C > 0$,

$$\mathbb{E}[Z] \leq C\sqrt{\log(n)}.$$

Hint: Break the integral into two parts, $[0, w)$ and $[w, \infty)$, for some judicious choice of $w > 0$; use a "trivial" bound for the first part, and use a bound that takes advantage of the lower integral limit for the second part.

(e) *Optional* (5 extra points). Let $X_1, X_2, \ldots$ be an infinite sequence of 1-subgaussian random variables (not necessarily independent), and let

$$Y := \max_{i=1,2,\ldots} \frac{|X_i|}{\sqrt{1 + \ln(i)}}.$$

Prove that, for some absolute constant $C > 0$,

$$\mathbb{E}[Y] \leq C.$$

(Of course, this result can also be used to prove that $\mathbb{E}[Z] \leq C\sqrt{\log(n)}$.)

Problem 4 (20 points)

Original PAC learning model: In the original definition of the PAC learning model, a learning algorithm does not get training data as input directly, but rather is provided access to an "Example Oracle" EX that, when queried, returns an independent draw from the probability distribution P over $\mathcal{X} \times \{0, 1\}$. The sample complexity of a learning algorithm is the number of times it queries EX. A learning algorithm is "efficient" if both its time complexity and its sample complexity are polynomial in d, $1/\epsilon$, $1/\delta$, and $\log |H|$; and a learning algorithm is "correct" if it returns a hypothesis $\hat{h}$ from H such that

$$\Pr\left[\Pr_{(X,Y)\sim P}[\hat{h}(X) \neq Y] \leq \epsilon\right] \geq 1 - \delta.$$

For a nice introduction to the original PAC model, please read 1.1–1.2 in Kearns & Vazirani. Note that we have elided some of the issues discussed there (like "representation size").

Two-oracle learning model: Now, consider a two-oracle model of learning, where a learning algorithm does not have access to EX as above, but instead has access to two other oracles, called the "Positive Example Oracle" EX_1 and the "Negative Example Oracle" EX_0. When EX_1 is queried, it returns an independent draw from P_1, which is the probability distribution P conditioned on the label being 1. Analogously, when EX_0 is queried, it returns an independent draw from P_0, which is the probability distribution P conditioned on the label being 0. In this learning model, a learning algorithm is "efficient" under the same criteria as in the original PAC model (though now sample complexity counts the number of queries to EX_1 and EX_0); and a learning algorithm is "correct" if it returns a hypothesis $\hat{h}$ from H such that

$$\Pr\left[\max\left\{\Pr_{(X,Y)\sim P_1}[\hat{h}(X) \neq Y], \Pr_{(X,Y)\sim P_0}[\hat{h}(X) \neq Y]\right\} \leq \epsilon\right] \geq 1 - \delta.$$

Your task: Prove that there is an efficient and correct learning algorithm in the original model if and only if there is an efficient and correct learning algorithm in the two-oracle model. You should assume that H contains, among possibly many other hypotheses, the "constant 1" hypothesis h_1 (where $h_1(x) = 1$ for all $x \in \mathcal{X}$) and the "constant 0" hypothesis h_0 (where $h_0(x) = 0$ for all $x \in \mathcal{X}$). Also, you should assume that P_1 and P_0 are well-defined; in particular, assume that $\Pr_{(X,Y)\sim P}[Y = 1]$ is neither zero nor one.

Problem 5 (20 points)

A decision list on $\{0,1\}^d$ is a function $h\colon \{0,1\}^d \to \{0,1\}$ of the following form:

- On input $\boldsymbol{x} \in \{0,1\}^d$:
  - If $c_1(\boldsymbol{x}) = 1$, then return b_1;
  - Else if $c_2(\boldsymbol{x}) = 1$, then return b_2;
  - $\vdots$
  - Else if $c_l(\boldsymbol{x}) = 1$, then return b_l;
  - Else return b_{l+1}.

Above, the function is parameterized by $((c_1, b_1), \ldots, (c_l, b_l), b_{l+1})$, where each c_i is a clause given by a single literal (i.e., $c_i(\boldsymbol{x}) = x_j$ or $c_i(\boldsymbol{x}) = 1 - x_j$ for some $j \in \{1, \ldots, d\}$), and each $b_i \in \{0,1\}$.

(a) Let DL be the hypothesis class of decision lists on $\{0,1\}^d$ (where the length l can be arbitrary). Prove that $|\mathsf{DL}| = O(3^{2d} d!)$.

(b) Let CONJ (respectively, DISJ) be the family of conjunctions (respectively, disjunctions) on $\{0,1\}^d$. Prove that $\mathsf{CONJ} \cup \mathsf{DISJ} \subseteq \mathsf{DL}$.

(c) *Optional* (5 extra points). Give an efficient algorithm that, on input $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) \in \{0,1\}^d \times \{0,1\}$ with the promise that there exists a decision list $c^* \in \mathsf{DL}$ such that $y_i = c^*(\boldsymbol{x}_i)$ for all i, outputs a decision list $h \in \mathsf{DL}$ that satisfies $h(\boldsymbol{x}_i) = y_i$ for all i. (The length of the decision list that the algorithm returns need not be the same as the length of c^*.) *Hint.* A greedy algorithm works here.