Homework 2

November 26, 2025

Instructions: Please submit a written solution to at least **one** problem below. You are welcome to submit more, but your grade will be the highest score for any single problem you submit.

Problem 1: **Fun and games with SoS**

(a) A special case of the celebrated hypercontractivity inequality for low-degree polynomials is the following: let f be a degree-k multilinear polynomial, that is, $f(x) = \sum_{S \subseteq [n]} c_S \prod_{i \in S} x_i$, and let U be the uniform distribution over $\{\pm 1\}^d$. Then, we have that

$$\mathop{\mathbb{E}}_{X \sim U}[f(X)^4] \leq 9^k \mathop{\mathbb{E}}_{X \sim U}[f(X)^2]^2 \, .$$

Prove that this inequality has a sum-of-squares proof in the coefficients of f. What is the degree of the proof?

(b) Let's explore the power of SoS for certifying injective norms of moment tensors, a canonical way in which SoS is often used in the algorithmic and statistical literature. There is a semi-formal sense in which all SoS is doing for this problem is finding the "best" flattening of the tensor into some matrix, and then applying a spectral bound on this matrix. We will show in this problem that this can be very powerful.

Given some distribution D, let

$$\widetilde{C}_4 = \max_{\widetilde{\mathbb{E}} \models_{4k} \|v\|_2^2 = 1} \widetilde{\mathbb{E}}[\langle v, X \rangle^k] \, .$$

Consider the matrix

$$M_4 = \mathbb{E}[(X \otimes X)(X \otimes X)^\top] \, .$$

This matrix is the "naive" flattening of the moment tensor T_4 into a $d^2 \times d^2$ matrix. Give a concrete example of a distribution where $\|M_4\|_\infty = \omega(1)$, but $\widetilde{C}_k = O(1)$.

(c) **Extra credit:** Devise an explicit distribution D so that $\sup_{\|v\|=1}[\langle v, X \rangle^4] = O(1)$, but $\widetilde{C}_k = \omega(1)$.

Problem 2: **Algorithmic primitives for MMW.** There were some missing details in implementing MMW for fast robust mean estimation. In this problem, your goal is to fill some of them out.

(a) Complete the proof of Lemma 2.1 in Lecture 14, which we restate here for completeness:

Lemma 0.1. *There is an (randomized) algorithm, which given τ_i satisfying*

$$\sum_{i \in S_{\text{good}}} \tau_i \leq \frac{1}{4} \sum_{i \in S} \tau_i \, ,$$

outputs a set of points S' so that with high probability, we've removed more bad points than good points, and $\sum_{i \in S'} \tau_i \leq \frac{2}{3} \sum_{i \in S} \tau_i$. Moreover, this algorithm runs in nearly linear time.

(b) Now we'll see some of the details of actually implementing the MMW update, specifically, the efficient implementation of the matrix exponential. First, devise a randomized algorithm for the following problem: Given a dataset of points $X_1, \ldots, X_n \in \mathbb{R}^d$, and let and let X be the matrix whose i-th row is X_i. For any parameter k, compute $\hat{s}_i$ so that $0.9 \cdot s_i \leq \hat{s}_i \leq 1.1 \cdot \hat{s}_i$, where

$$s_i = X_i^\top \left(X^T X\right)^k X_i \ ,$$

for all $i = 1, \ldots, n$. The algorithm should run in time $\widetilde{O}(ndk)$, where the $\widetilde{O}$ hides polylogarithmic factors in n, d, k. You may use the following fact, which is a slight strengthening of the standard Johnson-Lindenstrauss dimensionality reduction method, without proof:

Lemma 0.2. *Let $A, B \in \mathbb{R}^{d \times d}$ be symmetric, and suppose that $B = MM^\top$ for some symmetric matrix M. Let $\delta > 0$, and let $S \in \mathbb{R}^{r \times d}$ have entries which are independent copies of $\mathcal{N}(0, 1/r)$, where $r = O(\log(d/\delta)/\varepsilon^2)$. Then, with probability $1 - \delta$, it holds that*

$$\left|\left\langle A, MU^\top UM \right\rangle - \langle A, B \rangle\right| \leq \varepsilon \|A\|_\infty \cdot \mathrm{tr}(B) \ .$$

(c) Now, demonstrate that there is a degree $O(\log(1/\varepsilon))$ matrix polynomial P so that for any symmetric matrix M with spectral norm at most 1, we have that

$$(1 - \varepsilon)P(M) \preceq e^M \preceq (1 + \varepsilon)P(M) \ .$$

(d) Combine these two facts to demonstrate an efficient algorithm for computing the scores τ_i in equation (4) from Lecture 14.

Problem 3: **Fun and games with differential privacy.**

(a) A major difficulty with the techniques presented in this class are that they are not sensitive to the true scale of the dataset: rather they can only witness some *a priori* bound that we provide it. If instead we want the algorithm to be optimal with respect to the optimal *a posteriori* bound, this turns out to be much more challenging.

Consider the following problem: give a $(\varepsilon, 0)$-differentially private algorithm so that given a dataset of points $X_1, \ldots, X_n \sim \mathcal{N}(\mu, \sigma^2)$ for $\mu \in [-1, 1]$, and any $\sigma > 0$, outputs a $\widehat{\mu}$ so that with high probability, $|\mu - \widehat{\mu}| \leq 0.5 \cdot \sigma$. Note that non-privately, the empirical mean satisfies this with very high probability, for n sufficiently large. Show that there is no algorithm for this problem which succeeds with any finite number of samples.

(b) **Private medians.** Suppose $X = \{X_1, \ldots, X_n\}$ is a dataset over $\{1, \ldots, m\}$. Give a ε-DP algorithm that has the following utility guarantee. Let $\beta > 0$. For all $y \in R$, let $q(X, y)$ be the number of elements in X which are less than y. Then, the algorithm should output some point Y so that with probability $1 - \beta$, we have that

$$\left|q(X, y) - \frac{n}{2}\right| \leq O\left(\frac{\log m + \log 1/\beta}{\varepsilon}\right) \ .$$

(c) Now, assume you have data from $\mathcal{N}(\mu, 1)$ for some $|\mu| \leq R$, for some known parameter R. Generalize the previous algorithm to obtain a private median estimation algorithm for this setting. Conclude that this implies an algorithm for Gaussian mean estimation which is simultaneously robust and private. What are the quantitative guarantees you achieve, if the dataset is α-corrupted?