

# Efficient Mean Estimation with Pure Differential Privacy via a Sum-of-Squares Exponential Mechanism\*

Samuel B. Hopkins<sup>†</sup>      Gautam Kamath<sup>‡</sup>      Mahbod Majid<sup>§</sup>

November 25, 2021

## Abstract

We give the first polynomial-time algorithm to estimate the mean of a  $d$ -variate probability distribution with bounded covariance from  $\tilde{O}(d)$  independent samples subject to *pure* differential privacy. Prior algorithms for this problem either incur exponential running time, require  $\Omega(d^{1.5})$  samples, or satisfy only the weaker *concentrated* or *approximate* differential privacy conditions. In particular, all prior polynomial-time algorithms require  $d^{1+\Omega(1)}$  samples to guarantee small privacy loss with “cryptographically” high probability,  $1 - 2^{-d^{\Omega(1)}}$ , while our algorithm retains  $\tilde{O}(d)$  sample complexity even in this stringent setting.

Our main technique is a new approach to use the powerful *Sum of Squares method (SoS)* to design differentially private algorithms. *SoS proofs to algorithms* is a key theme in numerous recent works in high-dimensional algorithmic statistics – estimators which apparently require exponential running time but whose analysis can be captured by *low-degree Sum of Squares proofs* can be automatically turned into polynomial-time algorithms with the same provable guarantees. We demonstrate a similar *proofs to private algorithms* phenomenon: instances of the workhorse *exponential mechanism* which apparently require exponential time but which can be analyzed with low-degree SoS proofs can be automatically turned into polynomial-time differentially private algorithms. We prove a meta-theorem capturing this phenomenon, which we expect to be of broad use in private algorithm design.

Our techniques also draw new connections between differentially private and robust statistics in high dimensions. In particular, viewed through our proofs-to-private-algorithms lens, several well-studied SoS proofs from recent works in algorithmic robust statistics directly yield key components of our differentially private mean estimation algorithm.

---

\* Authors are in alphabetical order.

<sup>†</sup>UC Berkeley and MIT. [samhop@mit.edu](mailto:samhop@mit.edu). Supported by a Miller Postdoctoral Fellowship and a Simons Postdoctoral Fellowship. Part of this work was conducted while visiting the Simons Institute for the Theory of Computing.

<sup>‡</sup>Cheriton School of Computer Science, University of Waterloo. [g@csail.mit.edu](mailto:g@csail.mit.edu). Supported by an NSERC Discovery Grant and a University of Waterloo Startup grant.

<sup>§</sup>Cheriton School of Computer Science, University of Waterloo. [m2majid@uwaterloo.ca](mailto:m2majid@uwaterloo.ca). Supported by an NSERC Discovery Grant, a Graduate Excellence Award in Computer Science, a David R. Cheriton Graduate Scholarship, and a Waterloo CPI Cybersecurity and Privacy Excellence Graduate Scholarship.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Related Work . . . . .	4
<b>2</b>	<b>Techniques</b>	<b>5</b>
2.1	The SoS Exponential Mechanism . . . . .	5
2.2	Private Mean Estimation . . . . .	7
<b>3</b>	<b>Preliminaries</b>	<b>11</b>
3.1	SoS Proofs and Pseudoexpectations . . . . .	11
3.2	Privacy . . . . .	12
<b>4</b>	<b>Meta-Theorem on SoS Exponential Mechanism</b>	<b>13</b>
4.1	Proof of Theorem 4.5 . . . . .	15
4.2	Proofs of Lemmas . . . . .	16
<b>5</b>	<b>Coarse Estimation</b>	<b>18</b>
5.1	First Algorithm: Coordinate-Wise . . . . .	19
5.2	Second Algorithm: SoS Exponential Mechanism . . . . .	22
5.3	Robustness . . . . .	29
<b>6</b>	<b>Fine Estimation</b>	<b>29</b>
6.1	Preliminaries . . . . .	34
6.2	Main Algorithm and Theorem . . . . .	36
6.3	Halt Estimation . . . . .	37
6.4	Distance Estimation . . . . .	38
6.5	Gradient Estimation . . . . .	42
6.6	Gradient Descent . . . . .	52
6.7	Robustness . . . . .	53
<b>7</b>	<b>Lower Bounds</b>	<b>53</b>
<b>A</b>	<b>Deferred Proofs</b>	<b>61</b>
<b>B</b>	<b>Lemmata</b>	<b>63</b>

# 1 Introduction

*Mean estimation* is perhaps the most elementary statistical task: given samples from a probability distribution  $D$ , estimate its expected value. In this paper, we study mean estimation in  $d$  dimensions subject to *differential privacy (DP)* [DMNS06], a rigorous notion of data privacy. Privacy is of natural concern in high-dimensional statistics, where data may be sensitive and standard estimators like the empirical mean may leak information about individuals in a dataset (see, e.g., privacy attacks in [DN03, HSR<sup>+</sup>08, BUV14, DSS<sup>+</sup>15] and the survey [DSSU17]). On the other hand, privacy and statistical estimation seem largely compatible: in the large-sample limit, good statistical estimators will have vanishing dependence on each individual sample anyway. Indeed, even though the empirical mean (the natural benchmark for accuracy in mean estimation) is not differentially private, estimators are known which match its accuracy guarantees while also providing differential privacy [BKSU19, KSU20]. Namely, an accurate and private estimate of the mean can be obtained using  $n = O(d)$  samples. However, known estimators achieving this sample complexity require exponential running time.

If one instead focuses on polynomial-time algorithms, existing estimators all face significant drawbacks: either they require  $n \geq d^{1+\Omega(1)}$  samples, or they may leak private information with probability  $2^{-d^c}$  for small  $c > 0$  [HT10, KLSU19, KSU20]. (In privacy language, they satisfy only *concentrated* or *approximate* differential privacy.) Since a major goal of privacy is to make strong assurances on leakage of sensitive information, best practices disallow private data leakage even with “cryptographically” small probability  $2^{-\text{poly}(d)}$ . One way to satisfy this stringent requirement is to provide *pure* differential privacy, which disallows substantial leakage of private information with any nonzero probability. Thus, the main question in our paper is:

*Is there a polynomial-time pure DP algorithm for mean estimation using  $n = O(d)$  samples?*

Our main result answers this question affirmatively, up to logarithmic factors. To state our main result, we define differential privacy:

**Definition 1.1** ((Pure) differential privacy). *For  $\varepsilon > 0$ , a (randomized) algorithm  $A$  which takes  $n$  inputs  $X_1, \dots, X_n$  is  $\varepsilon$ -differentially private ( $\varepsilon$ -DP) if for every pair of inputs  $X_1, \dots, X_n$  and  $X'_1, \dots, X'_n$  such that  $X_i = X'_i$  for all except a single index  $i \in [n]$  and for all possible events  $S \subseteq \text{Range}(A)$ ,*

$$\mathbb{P}(A(X_1, \dots, X_n) \in S) \leq e^\varepsilon \cdot \mathbb{P}(A(X'_1, \dots, X'_n) \in S).$$

**Theorem 1.2** (Main Theorem — Combination of Theorem 5.1, Theorem 6.1). *For every  $n, d \in \mathbb{N}$  and  $R, \alpha, \varepsilon, \beta > 0$  there is a polynomial-time  $\varepsilon$ -DP algorithm such that for every distribution  $D$  on  $\mathbb{R}^d$  such that  $\|\mathbf{E}_{X \sim D} X\| \leq R$  and  $\text{Cov}_{X \sim D}(X) \preceq I$ , given  $X_1, \dots, X_n \sim D$ , with probability  $1 - \beta$  the algorithm outputs  $\hat{\mu}$  such that  $\|\hat{\mu} - \mathbf{E}_{X \sim D} X\| \leq \alpha$ , so long as*

$$n \geq \tilde{O} \left( \frac{d + \log(1/\beta)}{\alpha^2 \varepsilon} + \frac{d \log R + \min(d, \log R) \cdot \log(1/\beta)}{\varepsilon} \right).$$

*Furthermore, if an  $\eta$ -fraction of the samples  $X_1, \dots, X_n$  are adversarially corrupted, the algorithm maintains the same guarantee, at the cost that now  $\|\hat{\mu} - \mathbf{E} X\| \leq \alpha + O(\sqrt{\eta})$ .*

The sample complexity of our algorithm is nearly linear in  $d$ , thereby answering our main question up to logarithmic factors. Beyond this core goal, we highlight that our algorithm is *also* robust to adversarial contaminations and enjoys sub-Gaussian confidence intervals when  $n \gg \frac{d \log R + \min(d, \log R) \log(1/\beta)}{\varepsilon}$ , both sought-after features in recent non-private algorithms [DKK<sup>+</sup>19,

LRV16, Hop20, CFB19]. The sample complexity of our algorithm is nearly optimal, with the exception of the term  $(\log(1/\beta) \cdot \min(d, \log R))/\varepsilon$  – see Theorem 7.2 for corresponding lower bounds. (Put differently, information-theoretically it is possible to achieve sub-Gaussian confidence intervals under the slightly milder assumption  $n \gg \frac{d \log R + \log(1/\beta)}{\varepsilon}$ .)

**Beyond clip-and-noise** Obtaining the guarantees of our algorithm requires going beyond existing techniques for private mean estimation, which we briefly review. A common technique in differential privacy is to “just add noise.” More precisely, suppose that  $f(X_1, \dots, X_n)$  is a *bounded sensitivity* function of  $n$  inputs, meaning that replacing a single input  $X_i$  with an arbitrary  $X'_i$  changes the value of  $f$  by at most  $\Delta$  (in an appropriate choice of norm). Then,  $f(X_1, \dots, X_n) + Z$  will be private, where  $Z$  is an appropriate random variable whose magnitude depends on  $\Delta$ .

Existing polynomial-time algorithms for private mean estimation all take this approach. First, to limit the sensitivity of the empirical mean, the algorithms clip the samples  $X_1, \dots, X_n \in \mathbb{R}^d$  to lie in an  $\ell_2$  ball. Considering the case  $\text{Cov}(X) \approx I$ , we will have  $\|X - \mathbf{E}X\| \approx \sqrt{d}$ , so using a ball of radius at least  $\sqrt{d}$  is unavoidable without introducing too much error in the clipping phase. Then, the algorithms output  $\bar{\mu} + Z$ , where  $\bar{\mu}$  is the empirical mean of the clipped samples.

If one takes the coordinates of  $Z$  to be draws from a Laplace distribution, the resulting algorithm will satisfy  $\varepsilon$ -DP, but the relatively heavy tails of the Laplace distribution impose a cost that  $n$  must be at least  $d^{1.5}/\varepsilon$  to obtain nontrivial guarantees. On the other hand, if  $Z$  is Gaussian, the resulting algorithm appears to tolerate  $n \approx d/\varepsilon$  samples, but it no longer satisfies *pure* DP. Instead, to guarantee privacy loss at most  $\varepsilon$  with probability at least  $\delta$  over the internal randomness in the algorithm,  $n \geq d\sqrt{\log(1/\delta)}/\varepsilon$  is required (i.e. the algorithm satisfies *concentrated* DP), meaning that for “cryptographically small”  $\delta$ , this algorithm, too, has super-linear sample complexity.

One might naturally suspect, therefore, that strong privacy guarantees like this simply require  $\Omega(d^{1.5})$  samples. Indeed, if nastier distributions than we consider here (violating bounded covariance) are allowed, these two bounds of  $\Omega(d^{1.5})$  and  $\Omega(d)$  are known to be tight for pure and concentrated DP, respectively [HT10, BUV14, KLSU19], and similar separations were previously conjectured even for bounded-covariance distributions [BKSW19].

However, [KSU20] (building on related techniques of [BKSW19]) show that in exponential time one can go beyond the clip-and-noise approach to mean estimation. In particular, a tournament-based approach gives a *pure-DP* algorithm using  $O(d/\varepsilon)$  samples. We obtain nearly-matching guarantees in polynomial time – ours is the first polynomial-time private algorithm to go beyond the clip-and-noise approach.

**Sum-of-Squares Proofs to Private Algorithms** *Proofs to Algorithms* has become a powerful algorithm-design technique in computational high-dimensional statistics. Roughly speaking, the proofs to algorithms technique shows that statistical estimation problems which can be solved to a given accuracy by some (not necessarily polynomial-time) estimator can actually be solved in polynomial time with the same accuracy *if the analysis of that estimator can be captured by a certain restricted but powerful formal proof system, known as the SoS proof system*. This insight has had major consequences in robust and heavy-tailed statistics, clustering, learning latent variable models, and beyond. (For instance, [HL18, KS17b, KS17a, PS17], and the survey [RSS18].)

We show that this approach also applies to private algorithm design. Our techniques give a generic method to turn (potentially) exponential-time instances of the workhorse *exponential mechanism*, whose breadth of applicability it is hard to overstate [MT07], into polynomial time algorithms, when their analyses are captured by the same SoS proof system. This gives us the following *proofs-to-algorithms* principle for private algorithm design, which we anticipate will be

widely applicable:

**Proofs to Private Algorithms** (see Theorem 4.5): Instances of the exponential mechanism with low-degree SoS proofs of bounded sensitivity and utility automatically yield computationally-efficient private algorithms.

(See Section 2 for a description of the exponential mechanism and more on SoS proofs.) We are able to capture this principle as a formal meta-theorem, Theorem 4.5 – since its statement requires a few more technical definitions than we are ready to state, we defer it for now.

**Convex Programming and Private Algorithms** Like nearly all applications of the Sum of Squares method, algorithms which result from our proofs-to-private-algorithms approach ultimately use convex programs – semidefinite programs, in our case. Our techniques give a substantially novel approach to convex programming in private algorithms. Prior works largely do this in one of two ways. Some privatize the input *before* applying a convex program to it (e.g., privatizing a graph and then post-processing with convex programming [BBDS12, EKKL20]). This can be limiting, as the convex program cannot itself be helpful in achieving privacy, only in some post-processing. Other algorithms use a private solver for the convex program itself [HRRU14, LKKO21]. This can be quite technically challenging: few DP solvers for convex programs are known, and in particular, we are not aware of any generic DP algorithm for solving semidefinite programs.

Instead, our algorithms *hide convex programs behind the exponential mechanism*. Namely, we use convex programs as score functions. (Again, see Section 2 for an explanation of this terminology.) Our key insight is: *when convex programs are married to the exponential mechanism in the correct way, the resulting exponential mechanisms can be implemented in polynomial time using private log-concave sampling algorithms*. This builds directly on [BST14], who use a similar approach but for much simpler un-constrained convex optimization problems resulting from empirical risk minimization problems. The analysis of this interplay of exponential mechanism and convex programming is completely generic – meaning it has nothing to do with the particular setting of mean estimation – and is captured in the proof of our (meta)-Theorem 4.5.

**Robust Statistics and Privacy** Robust statistics, the study of statistics in the presence of corrupted samples, has enjoyed a recent renaissance [DK19]. Robustness and privacy are at least spiritual cousins – both demand that a statistical estimator not change its behavior “too much” when one or several samples are replaced arbitrarily. However, the formal requirements for robustness and privacy are rather different. The output of a good robust mean estimator should not move far in Euclidean distance when 1% of the samples it is given are corrupted by a malicious adversary. Privacy, by contrast, demands that the *distribution* of outputs of an algorithm not shift too much when any single sample is replaced by another.

In spite of this formal difference, [DL09] was able to use robust estimators in one dimension to construct private estimators. This suggests that the flurry of recent algorithms for high-dimensional robust statistics with strong provable guarantees should yield high-dimensional private estimators.

Our work makes good on this promise. In particular, our algorithm for private mean estimation ultimately employs a well-studied Sum-of-Squares SDP from robust mean estimation, and our analysis applies several SoS proofs originally formulated to analyze that SDP in robust statistics. Thus, our proofs-to-private-algorithms approach gives a lens through which algorithms from robust statistics can yield private algorithms.

Lastly, there is an even more direct connection between our particular result and recent developments in robust statistics. The robustness renaissance was kicked off by [DKK<sup>+</sup>16, LRV16],

Algorithm	Sample Complexity	Privacy	Sub-Gaussian Rates	Robust	Poly Time
Sample Mean	$d$	None	No	No	Yes
[KSU20]-A	$d$	Pure	-	Yes	No
[KSU20]-B	$d$	Concentrated	No	No	Yes
[KSU20]-C	$d^{1.5}$	Pure	No	No	Yes
[LKKO21]-A	$d$	Approximate	-	Yes	No
[LKKO21]-B	$d^{1.5}$	Concentrated	No	Yes	Yes
Theorem 1.2	$d$	Pure	$n \gg \frac{d \log R + \min(d, \log R) \log(1/\beta)}{\epsilon}$	Yes	Yes

Table 1: Comparing private mean estimation algorithms for distributions with bounded covariance. (Some papers contain several algorithms.) Sample complexity column ignores logarithmic factors in  $d$ . “-” indicates that we are not aware of any analysis in the literature.

which gave the first polynomial-time algorithms for robustly learning a Gaussian where corrupted samples can be tolerated with dimension-independent error. Prior polynomial-time algorithms for the same problem have guarantees no better than what can be obtained by naive sample-clipping, and as a result incur error scaling as  $\eta \cdot \text{poly}(d)$  when an  $\eta$ -fraction of samples are corrupted. Just as [DKK<sup>+</sup>16, LRV16] gave the first algorithms with guarantees going beyond naive sample clipping in the robust setting, our work is the first to go beyond clip-and-noise in the private setting.

## 1.1 Related Work

A mature body of work focuses on private mean estimation. The most relevant results restrict the underlying distribution to satisfy a moment bound (including sub-Gaussianity), examples include [BD14, KV18, BKSW19, BS19, KLSU19, CWZ19, KSU20, WDX20, DFM<sup>+</sup>20, BDKU20, AAK21, BGS<sup>+</sup>21, KLZ21, HLY21, KMS<sup>+</sup>21, AL21]. All prior study of the multivariate case either focuses on concentrated or approximate DP, or provides computationally-inefficient algorithms for pure DP. We are the first to give an efficient  $O(d)$ -sample algorithm for multivariate mean estimation under pure DP. This matches the sample complexity of the best known algorithms under concentrated DP, while simultaneously strengthening the privacy guarantee.

Other prior work studies private mean estimation of arbitrary (bounded) distributions [BUV14, SU15, DSS<sup>+</sup>15]. Interestingly, in this case, the optimal sample complexity under pure and concentrated DP are separated by a factor of  $\sqrt{d}$ , leading to the naive Laplace mechanism being effectively optimal. It appears that our bounded moments assumption induces a qualitatively different structure, which eliminates the benefits of relaxing to concentrated DP. Pinpointing the precise conditions under which a separation occurs remains an interesting direction for future work.

Other works study private statistical estimation in different and more general settings, including mixtures of Gaussians [KSSU19, AAL21], graphical models [ZKKW20], discrete distributions [DHS15], and median estimation [AMB19, TVGZ20]. Some recent directions involve guaranteeing user-level privacy [LSY<sup>+</sup>20, LSA<sup>+</sup>21], or a combination of local and central DP for different users [ADK19b]. See [KU20] for further coverage of DP statistical estimation.

Several other works employ sampling-based methods to efficiently implement the exponential mechanism [KT13, BST14, ADK<sup>+</sup>19a, LMV21]. [KT13, ADK<sup>+</sup>19a, LMV21] construct sophisticated sampling algorithms by hand to sample from a non-log-concave distributions; by contrast, we use convex programming to construct our score functions, ensuring that we always stay within the realm of log-concave sampling algorithms. [BST14] constructs the private log-concave sampler we use in our work, but they employ it only to sample from comparatively simple log-concave distri-

butions arising from empirical risk minimization of convex loss functions. A recent work of [MV21] provides a faster algorithm for private log-concave sampling.

Another recent line of work studies sub-Gaussian confidence intervals for mean estimation of heavy-tailed distributions (i.e., assuming only bounded covariance). Lugosi and Mendelson [LM19b] proposed an inefficient algorithm, with an SoS-based algorithm coming in [Hop20]; see also the survey [LM19a]. Some works considered efficient algorithms for simultaneously robust and heavy-tailed mean estimation [DL19, PBR19, DKP20]. A recent result shows that the core solution concepts for robust and heavy-tailed mean estimation can be considered equivalent [HLZ20]. Our work demonstrates that the line of efficient estimators inspired by [Hop20] is simultaneously effective for robust, heavy-tailed, and private mean estimation.

Relatively limited work simultaneously considers privacy and the other constraints of robustness and heavy-tailed estimation. The main result is [LKKO21] which considers robust and private mean estimation. Our result can be seen as improving on the running time or sample complexity of their two algorithms, and the privacy guarantee of both. While other prior works focus on heavy-tailed distributions [BD14, KSU20, WXDX20, KLZ21], these do not address the primary goal in the non-private setting, which is to achieve sub-Gaussian rates with respect to the error probability. Instead, they generally prove guarantees with constant probability of success, which can be boosted at the cost of a multiplicative (rather than additive) logarithmic factor which is inverse in the failure probability. Note that some of the previous (inefficient) cover- and median-based approaches to private estimation [BKSU19, KSU20, LKKO21, BGS<sup>+</sup>21, AAK21] have varying levels of robustness and sub-Gaussian rates for heavy-tailed settings. However, none of their results give computationally efficient estimators for pure DP. A simultaneous and independent work of [LKO21] demonstrates an interesting connection between resilience [SCV18] and private estimation. They exploit this connection to design robust and private algorithms for a variety of settings, including mean estimation, covariance estimation, PCA, and more. However, their focus is on providing inefficient algorithms under the constraint of approximate DP, while our goal is to give a framework for efficient algorithms under pure DP. For a summarized comparison of our work with recent private algorithms, see Table 1.

## 2 Techniques

### 2.1 The SoS Exponential Mechanism

Before we turn to our algorithm for mean estimation, we offer a little more detail on the proofs-to-private-algorithms approach and its core component, which we call the SoS exponential mechanism. We begin with a review of the exponential mechanism itself.

**The Exponential Mechanism** Consider the general problem of privately selecting one among a set of candidates  $\mathcal{C}$  given a dataset  $X$ , where the quality of a candidate  $x \in \mathcal{C}$  depends on the dataset  $X$ . The candidates  $\mathcal{C}$  could represent many different things depending on the context. In a statistical setting one may often think of  $\mathcal{C}$  as a class of probability distributions,  $X$  as a list of samples from one of those distributions, and the goal is to select the distribution from which  $X$  came (up to small error).

To apply the exponential mechanism for this problem, one first finds a *score function*  $s(X, x)$  which assigns a (real-valued) score to each dataset-candidate pair, ideally such that  $x$ 's which are "good" for a given  $X$  receive high scores. Given  $X$  and a privacy parameter  $\varepsilon > 0$ , the exponential mechanism will sample a random  $x$  with probabilities  $\mathbb{P}(x) \propto \exp(\varepsilon \cdot s(X, x))$ . The output is  $\varepsilon$ -

differentially private so long as the score function satisfies a *bounded sensitivity* property: for any pair of neighboring datasets  $X, X'$ <sup>1</sup> and for all  $x \in \mathcal{C}$ , one has  $|s(X, x) - s(X', x)| \leq 1$ .

Beyond privacy, one also wants the resulting  $x$  to be useful. While this can also mean different things depending on the context, we will take “utility” to mean that  $x$  is close to some good  $x^*(X)$ . To prove that this happens for the exponential mechanism defined above, one shows:

1. High-scoring  $x$ 's are good: if  $s(X, x) \geq 0$ , then  $\|x - x^*\| \leq \alpha$ , for a small  $\alpha > 0$ . (The choice of 0 is without loss of generality; the mechanism is invariant under additive shifts of  $s$ .)
2. Not too few high-scoring  $x$ 's:  $\frac{|\{x \in \mathcal{C} : s(X, x) \geq t\}|}{|\mathcal{C}|} \geq \frac{1}{r}$ . (If  $\mathcal{C}$  is an infinite set one can replace  $|\cdot|$  with some measure of volume.)

Then a simple argument shows that  $x$  such that  $\|x - x^*\| \leq \alpha$  is selected with probability at least  $1 - r \exp(-\varepsilon t)$ . In other words, the mechanism selects a good  $x$  so long as  $t \gg \frac{\log r}{\varepsilon}$ .

The exponential mechanism can lead to computationally inefficient algorithms for two basic reasons. First, in high-dimensional statistics it is often natural to use score functions which seem hard to compute – the Tukey depth, for just one example [LKKO21, BGS<sup>+</sup>21]. Second, as we will face when we turn to mean estimation, even with score functions that are easy to compute, sampling  $x$  with  $\mathbb{P}(x) \propto \exp(\varepsilon \cdot s(X, x))$  may not be computationally tractable.

**Convex programs as score functions** The SoS exponential mechanism can address both of these sources of intractability for instances of the exponential mechanism where the *proofs* of bounded sensitivity are captured in a powerful formal proof system, the *Sum of Squares proof system* (SoS). To de-mystify this a little without yet delving into the details of SoS proofs, we can see a high-level picture of the algorithms which use the SoS exponential mechanism. Ultimately, these algorithms fit into the exponential mechanism framework, but with special score functions.

To wit, suppose that we can arrange for the score function  $s(X, x)$  to take the form of a linear optimization problem over a convex set  $\mathcal{K}(X)$  with an additional linear constraint involving  $x$ :

$$s(X, x) = \max_{y \in \mathcal{K}(X)} \langle c, y \rangle \text{ such that } Ay = x$$

for some matrix  $A$  and vector  $c$ . As long as  $\mathcal{K}(X)$  admits a computationally efficient separation oracle,  $s(X, x)$  will be polynomial-time computable – this already addresses the first source of potential intractability in the exponential mechanism. A simple argument shows even more: for each  $X$ ,  $s(X, x)$  is actually concave in  $x$ . Thus, the distribution  $\mathbb{P}(x) \propto \exp(\varepsilon \cdot s(X, x))$  is log-concave, so we can (usually) sample from it in polynomial time! This sampling step itself has to be done privately: luckily for us, log-concave sampling is so well understood that private methods for polynomial-time log-concave sampling are known [BST14].

The restriction that the score function take the form of an optimization problem is not a major one: many useful score functions in high-dimensional statistics, such as the Tukey depth, are naturally expressible in this way. But how to find a score function which is simultaneously a *convex* optimization problem and maintains both bounded sensitivity and utility? The SoS method gives us a way to construct such a score function automatically, starting from any score function which is expressible as a (potentially non-convex) optimization problem, and for which the *proofs* of bounded sensitivity and utility are expressible in the SoS proof system.

---

<sup>1</sup>We say  $X, X'$  are neighboring if they differ on the presence/absence of just one individual.



## 2.2 Private Mean Estimation

We proceed with a high-level description of our algorithm for private mean estimation and its analysis. One could obtain our algorithm as an instance of proofs-to-private-algorithms, by:

- Constructing a certain simple-to-analyze exponential mechanism-based mean estimator.
- Writing the simple analysis as a series of SoS proofs and applying (meta)-Theorem 4.5.

However, so that our algorithm can be understood without tackling the SoS exponential mechanism in full generality, we will now give a more concrete description of the algorithm and its analysis. We give a high level version of this description here, and in the main body of our paper we actually provide a full analysis of the algorithm without appealing to Theorem 4.5.

Let us recall the setup for our problem. There is a random variable  $X$  on  $\mathbb{R}^d$  with  $\text{Cov}(X) \preceq I$  and  $\|\mathbf{E}X\| \leq R$ . The goal is to estimate  $\mathbf{E}X$  from i.i.d. copies  $X_1, \dots, X_n$ , subject to  $\varepsilon$ -DP. The promise  $\|\mathbf{E}X\| \leq R$  is information-theoretically necessary for pure differential privacy [KV18, BKSW19].

Our algorithm has strong guarantees in the presence of adversarially-corrupted samples, and obtains sub-Gaussian confidence intervals given heavy-tailed samples. In fact, this is a side-effect of the fact that our algorithm uses well-studied SDPs from the robust statistics setting, viewing them through the SoS exponential mechanism lens to obtain privacy. In particular, folklore adaptations of the analyses in [CFB19, KS17b] directly show that our algorithm is robust, so to avoid a proliferation of notation we give the proof in the non-robust setting – see Remarks 6.40 and 5.25.

Like prior algorithms for private mean estimation, our algorithm has two phases.

1. Reduce  $R$  to  $\text{poly}(d)$ , roughly by finding a large ball containing a large number of samples.
2. Estimate  $\mathbf{E}X$  under the assumption  $\|\mathbf{E}X\| \leq \text{poly}(d)$ .

We start with the second step, which captures much of our conceptual contribution – even under the assumption  $R \leq \sqrt{d}$ , prior algorithms could not achieve pure differential privacy. Then we discuss the first step, where prior algorithms require  $\Omega(d(\log R + \log(1/\beta)))$  samples. We give an algorithm with sample complexity  $\log R(d + \log(1/\beta))$ . When  $\log R \ll d$ , our algorithm improves over prior work. (Information-theoretically,  $d \log R + \log(1/\beta)$  is possible.)

### 2.2.1 Private mean estimation when $\|\mathbf{E}X\| \leq \text{poly}(d)$

Let us write  $\mu = \mathbf{E}X$ . For simplicity, for now we focus on the case that our goal is to find  $\hat{\mu}$  such that  $\|\hat{\mu} - \mu\| \leq O(1)$  using  $O(d)$  samples. (We can reduce from the  $\alpha$ -error case to this one by placing the samples in buckets containing around  $1/\alpha^2$  samples and taking sample means within each bucket.) We will also think of  $\varepsilon$  as a small constant. A merit of our approach is that it allows powerful techniques from robust statistics to be used for private algorithm design: in this case, our algorithm draws heavily on a robust mean estimation algorithm due to [CFB19], using SoS exponential mechanism to privatize its use of convex programming.

**Iterative refinement/gradient descent** Our algorithm will iteratively refine an initial estimate of the mean (without loss of generality, the origin), producing a series of estimates  $\hat{\mu}_0 = 0, \hat{\mu}_1, \dots, \hat{\mu}_T$ . In each step  $t$ , we will privately find a unit vector  $v$  such that  $\langle v, \mu - \hat{\mu}_t \rangle \geq \Omega(1) \cdot \|\mu - \hat{\mu}_t\|$ . Then we can replace  $\hat{\mu}_t$  with  $\hat{\mu}_{t+1} = \hat{\mu}_t + r \cdot v$  for some appropriate step size  $r > 0$ . By standard reasoning this means that  $O(\log d)$  steps suffice to obtain  $\|\hat{\mu}_T - \mu\| \leq O(1)$ .

This gradient-descent approach introduces only logarithmic overheads into our sample complexity and running time, so now we turn to the heart of our algorithm: privately finding  $v$ .

**Finding private gradients** Given samples  $X_1, \dots, X_n \in \mathbb{R}^d$  and  $\hat{\mu}_t$ , we would like to privately select a unit vector  $v$  such that  $\langle v, \hat{\mu}_t - \mu \rangle \geq \Omega(1) \cdot \|\hat{\mu}_t - \mu\|$ . We will use the exponential mechanism, for which we need to define a score function. For this, we are inspired by recent work in robust and heavy-tailed statistics, where the following has become a standard fact [LM19b]:

**Fact 2.1** (Directions with many outliers are good, informal). *Suppose that  $\|\hat{\mu}_t - \mu\| \gg 1$ . Then, with high probability over  $X_1, \dots, X_n$ , so long as  $n \gg d$ , there are at least  $0.9n$  samples  $X_i$  such that  $\langle X_i - \hat{\mu}_t, \frac{\mu - \hat{\mu}_t}{\|\mu - \hat{\mu}_t\|} \rangle \geq \Omega(1) \cdot \|\hat{\mu}_t - \mu\|$ . Furthermore, for every unit vector  $v$  such that*

$$|\{i : \langle X_i - \hat{\mu}_t, v \rangle \geq \Omega(1) \cdot \|\hat{\mu}_t - \mu\|\}| \geq 0.8n,$$

we have  $\langle v, \mu - \hat{\mu}_t \rangle \geq \Omega(1) \cdot \|\mu - \hat{\mu}_t\|$ .

Fact 2.1 makes a good choice of score function clear: we should use

$$s(X, v) = |\{i : \langle X_i - \hat{\mu}_t, v \rangle \geq \Omega(1) \cdot \|\hat{\mu}_t - \mu\|\}|.$$

Utility of this score function is captured by Fact 2.1, and bounded sensitivity is clear by construction. (In fact, it is exactly this bounded sensitivity property which has already made Fact 2.1 so important in robust and heavy-tailed statistics.) We do not necessarily know the value of  $\|\hat{\mu}_t - \mu\|$ , but getting a private estimate of this quantity is not too difficult – we privatize a procedure due to [CFB19].

A straightforward analysis shows that, since the volume of the  $d$ -dimensional unit ball is roughly  $\exp(d)$ , exponential mechanism with this score function will  $\varepsilon$ -privately select a good  $v$  so long as  $n \gg d/\varepsilon$ , which is exactly the sample complexity we expect for estimating  $\mu$  to error  $O(1)$ .

**Finding private gradients in polynomial time** Of course, the key problem is that sampling with  $\mathbb{P}(v) \propto \exp(\varepsilon \cdot s(X, v))$  may not be possible in polynomial time. Indeed, a seemingly-easier problem, finding the highest-scoring  $v$ , seems closely related to computing Tukey depth, which is NP-hard.

However, some hope comes again from robust/heavy-tailed statistics, where approximation algorithms (often based on semidefinite programming) for the problem of finding the highest scoring  $v$  have become an invaluable tool. Our starting point is the by-now standard construction of such a semidefinite relaxation, which we briefly review. First, we write this optimization problem as a degree-2 polynomial optimization problem in variables  $v_1, \dots, v_d$  and  $b_1, \dots, b_n$ , where the latter are constrained so as to be 0/1 indicators of  $\langle X_i - \hat{\mu}_t, v \rangle \geq r$  for some threshold value  $r$ :

$$\max \sum_{b=1}^n b_i \text{ such that } \|v\|^2 \leq 1, b_i^2 = b_i, \text{ and } b_i \langle X_i - \hat{\mu}_t, v \rangle \geq b_i \cdot r \text{ for all } i.$$

The degree-2 SoS relaxation of this optimization problem optimizes over (degree 2) *pseudoexpectations*, which are linear functionals  $\tilde{\mathbf{E}} : \mathbb{R}[b, v]_{\leq 2} \rightarrow \mathbb{R}$  defined on degree at most 2 polynomials in  $b, v$  which are normalized and positive:  $\tilde{\mathbf{E}} 1 = 1$  and  $\tilde{\mathbf{E}} p(b, v)^2 \geq 1$  for all linear  $p$ . Concretely:

$$\max_{\tilde{\mathbf{E}}} \sum_{i \leq n} b_i \text{ s.t. } \tilde{\mathbf{E}} \|v\|^2 \leq 1, \tilde{\mathbf{E}} b_i^2 = \tilde{\mathbf{E}} b_i, \text{ and } \tilde{\mathbf{E}} b_i \langle X_i - \hat{\mu}_t, v \rangle \geq r \cdot \tilde{\mathbf{E}} b_i \text{ for all } i. \quad (1)$$

This optimization problem can be solved via semidefinite programming – the resulting (equivalent) SDP optimizes over  $(1 + n + d) \times (1 + n + d)$  block matrices

$$\max \text{Tr } B \text{ such that } \begin{pmatrix} 1 & \tilde{b}^\top & \tilde{v}^\top \\ \tilde{b} & B & W^\top \\ \tilde{v} & W & V \end{pmatrix} \succeq 0, \text{Tr } V \leq 1, B_{ii} = \tilde{b}_i, \text{ and } \langle X_i - \hat{\mu}_t, W_i \rangle \geq r \cdot B_{ii}$$

Here,  $V$  is a proxy for the rank-one matrix  $vv^\top$  and similarly for  $B$  and  $bb^\top$ . This SDP is known to be a good approximation to the problem of finding the maximum-score  $v$ .

We prove the following fact. (As an aside, this is where SoS proofs enter the picture: establishing facts such as the below for SoS SDP relaxations in general requires constructing SoS proofs.)

**Fact 2.2** (Bounded Sensitivity and Utility for (1), informal). ***Bounded Sensitivity:** Changing a single sample in optimization problem (1) can change the objective value by at most 1. **Utility:** With high probability over  $X_1, \dots, X_n$ , if  $\|\hat{\mu}_t - \mu\| \gg 1$ , then any feasible  $\tilde{\mathbf{E}}$  in (1) with objective value at least  $0.8n$  satisfies  $\langle \tilde{\mathbf{E}}v, \mu - \hat{\mu}_t \rangle \geq \Omega(1) \cdot \|\mu - \hat{\mu}_t\|$ .*

While this proof largely adapts similar arguments in the robust statistics literature, there is a key technical innovation. Prior algorithms employing the SDP described above use a nontrivial rounding step to extract a good vector  $v$  from the  $d \times d$  PSD matrix  $V$ . However, for reasons we discuss below, to use the SoS exponential mechanism, it is important that  $v$  can be read directly off of the SDP (more precisely, that the rounding algorithm used is linear). This means we need a stronger rounding procedure than that used in prior works – we are able to show that  $\tilde{v} = \tilde{\mathbf{E}}v$ , a simple linear function of  $\tilde{\mathbf{E}}$ , is a good choice.

**Sampling with convex programs** The “utility” part Fact 2.2 solves the problem of finding a high-scoring  $v$  in polynomial time, via semidefinite programming. But we want to find such a high-scoring  $v$  privately.

The key idea is to use the SDP to construct a score function: *convexity of the set of feasible solutions and linearity of the objective function now imply log-concavity of the resulting sampling problem!* This observation, while simple, is remarkably powerful: it allows us to employ a well-studied SDP from robust statistics nearly out-of-the-box to obtain strong privacy guarantees.

Ultimately, we employ the score function:

$$s(X, v_0) = \max_{\tilde{\mathbf{E}}} \tilde{\mathbf{E}} \sum_{i \leq n} b_i \text{ s.t. } \tilde{\mathbf{E}} \|v\|^2 \leq 1, \tilde{\mathbf{E}} b_i^2 = \tilde{\mathbf{E}} b_i, \text{ and } \tilde{\mathbf{E}} b_i \langle X_i - \hat{\mu}_t, v \rangle \geq r \cdot \tilde{\mathbf{E}} b_i \text{ for all } i \text{ and } \tilde{\mathbf{E}} v = v_0,$$

together with the private sampling algorithm of [BST14], to sample from  $\mathbb{P}(v_0) \propto \exp(\varepsilon \cdot s(X, v_0))$ .

We make a few remarks on the precise way that  $s(X, v_0)$  depends on  $v_0$ ; that is, via the constraint  $\tilde{\mathbf{E}}v = v_0$ , since this choice is not accidental. First of all, it is important that the constraints of the optimization problem defining  $s(X, v_0)$  depend *linearly* on  $v_0$ ; otherwise we might not retain log-concavity of the resulting sampling problem. We can do this only because the rounding algorithm described in Fact 2.2, which proves that  $\tilde{\mathbf{E}}v$  itself is useful, is a simple linear function of  $\tilde{\mathbf{E}}$ .

We also note an important interplay between the “lifted” nature of the SDP and the utility analysis of the exponential mechanism. On the one hand, the power of the SDP comes from lifting from the  $d + n$  variables  $v, b$  to  $(d + n + 1)^2$  variables, and solving a convex problem in the lifted space. On the other hand, for the exponential mechanism to satisfy utility with just  $O(d)$  samples, it is important that we use it to sample in  $d$  dimensions (where, in particular, there is a  $2^{O(d)}$ -sized cover) rather than, say,  $d^2$  dimensions. This tension is resolved by hiding the additional variables inside the optimization problem which defines  $s$ , so that exponential mechanism still samples from a  $d$ -dimensional distribution, but we can still use the power of SoS and semidefinite programming.

**Lipschitzness** The sampling algorithm of [BST14] (like other algorithms for sampling from log-concave probability distributions) runs in polynomial time in the ambient dimension, *so long as the distribution has Lipschitz log-probabilities*. Natural approaches to force  $s(X, v_0)$  to be Lipschitz, which generally take the form of randomized smoothing, risk violating pure DP, because

an algorithm computing a randomized smoothing of  $s$  will have some small probability of quietly failing, at which point privacy is at risk.

To ensure that the resulting algorithm runs in polynomial time, therefore, we have to show that  $s(X, v_0)$  is Lipschitz with respect to  $v_0$ . To establish Lipschitzness, we make a two-step argument:

1. We show that dual solutions to the SDP defining  $s(X, v_0)$  have optimal dual certificates which are not too large in norm. (At most, say,  $\text{poly}(d, n)$ .)
2. We show that any such dual solution to the SDP for  $s(X, v_0)$  which certifies an upper bound of  $c$  can be adapted to a dual solution for  $s(X, v_0 + \Delta)$  which certifies an upper bound of  $c + \|\Delta\| \text{poly}(d, n)$ , for any small perturbation vector  $\Delta$ .

Together, these imply that  $s(X, v_0)$  is  $\text{poly}(d, n)$ -Lipschitz with respect to  $v_0$ . Since the Lipschitz constant only arises in our algorithm's running time, this suffices for our purposes. This concludes our overview of the second phase of our private mean estimation algorithm.

### 2.2.2 Coarse estimation: from $R$ to $\text{poly}(d)$

We now describe our algorithm to privately localize  $\mu = \mathbf{E} X$  to a ball of radius  $\text{poly}(d)$ , beginning only with the promise that  $\|\mathbf{E} X\| \leq R$  for some large number  $R$ . The goal is to do so using as few samples as possible. Existing efficient algorithms [KLSU19] can perform this task with probability  $1 - \beta$  using  $O(d(\log R + \log(1/\beta))/\varepsilon)$  samples (we present this analysis in our paper to capture the dependence on  $\log(1/\beta)$ ). If we used this algorithm, we would obtain sub-Gaussian confidence intervals only when  $n \gg \frac{d \log R + d \log(1/\beta)}{\varepsilon}$ . However, if we do not worry about running time, the same task can be accomplished using the exponential mechanism using only  $O(d \log R/\varepsilon + \log(1/\beta)/\varepsilon)$  samples – much fewer for  $\beta \ll 1$ , which is important for constructing confidence intervals.

While we do not quite obtain optimal complexity, we are able to improve on existing algorithms in the regime  $\log R \ll d$ ; our algorithm requires  $\tilde{O}(d \log R/\varepsilon + \log R \log(1/\beta)/\varepsilon)$  samples.

Our algorithm again follows the proofs-to-private-algorithms approach. We start with the following basic instantiation of the exponential mechanism. We are given samples  $X_1, \dots, X_n$ . With probability at least  $1 - \beta$  over the choice of  $X_1, \dots, X_n$ , as  $n \gg \log(1/\beta)$ , any ball of radius  $\text{poly}(d)$  containing  $0.9n$  of the samples will have center which has distance at most  $\text{poly}(d)$  to  $\mu$ . So, we would like to use the exponential mechanism with score function  $s(X, x) = |\{i : \|X_i - x\| \leq \text{poly}(d)\}|$  to select a point from the ball of radius  $R$  centered at the origin.

As before, it is not clear how to perform the sampling task that this would require in polynomial time. So, we replace the score function with a convex relaxation, in this case built from the “degree-4” SoS relaxation of of the following polynomial optimization problem:

$$\max_{b, x} \sum_{i=1}^n b_i \text{ s.t. } \|x\|^2 \leq R^2, b_i^2 = b_i, \text{ and } b_i \|X_i - x\|^2 \leq \text{poly}(d) \cdot b_i \text{ for all } i.$$

(The degree-4 SoS relaxation shows up here because the optimization problem involves polynomials of degree 3, and SoS relaxations are defined only for even degrees.) That is, we use the score function

$$s(X, x_0) = \max_{\tilde{\mathbf{E}}} \tilde{\mathbf{E}} \sum_{i=1}^n b_i \text{ s.t. } \tilde{\mathbf{E}} \text{ satisfies } \|x\|^2 \leq R^2, b_i^2 = b_i, b_i \|X_i - x\|^2 \leq b_i (R/10)^2, \text{ and } \tilde{\mathbf{E}} x = x_0$$

(for the definition of “satisfies”, see Section 3).

Once we have decided to use this particular SoS relaxation, the outlines of the algorithm and its analysis are largely similar to the second phase of the algorithm, but with different SoS proofs

plugged in. In particular, we prove an analogue of Fact 2.2 for this setting, and establish Lipschitzness of the SDP, so that we can use the same strategy as in second phase.

The key step is to show that a high-scoring  $x_0$  is has distance at most  $R/2$  to any  $x'$  which has distance  $\text{poly}(d)$  to  $0.8n$  samples. This statement turns out to have a relatively simple SoS proof. This shows that SoS exponential mechanism manages to reduce  $R$  to  $R/2$ . Iterating this  $\log R$  times completes the algorithm.

### 3 Preliminaries

#### 3.1 SoS Proofs and Pseudoexpectations

We give a brief overview of SoS; for details see [BS16].

**SoS Proofs** We first informally review the SoS proof system. Let  $p(x), p_1(x), \dots, p_m(x)$  be multivariate polynomials in indeterminates  $x_1, \dots, x_n$ . The SoS proof system can prove statements of the form:

$$\text{For all } x \in \mathbb{R}^n, \text{ if } p_1(x) \geq 0, \dots, p_m(x) \geq 0, \text{ then } p(x) \geq 0.$$

Polynomials are highly expressive, so a wide range mathematical statements can be encoded in the above form. An SoS proof of such a statement is a family of polynomials  $\{q_S(x) : S \subseteq [m]\}$  such that each  $q_S(x) = \sum_{i=1}^r (q_S^{(i)}(x))^2$  is a sum of squares, and  $p(x) = \sum_{S \subseteq [m]} q_S(x) \cdot \prod_{i \in S} p_i(x)$ . The proof has degree  $D \in \mathbb{N}$  if each term in this sum is a polynomial of degree at most  $D$ . We write:

$$p_1 \geq 0, \dots, p_m \geq 0 \vdash_D p \geq 0.$$

Additionally, we will need one non-standard definition: for any given  $j \in [m]$ , we say that an SoS proof is degree  $D_j$  with respect to  $p_j$  if every term  $q_S(x) \cdot \prod_{i \in S} p_i(x)$  in the proof such that  $j \in S$  has degree at most  $D'$ . We will sometimes write  $p_1 \geq 0, \dots, p_m \geq 0 \vdash_{D, \deg_{p_j} = D_j} p \geq 0$ .

SoS proofs are dual solutions to semidefinite programs arising from the Sum of Squares method, where pseudoexpectations are primal solutions. As in many applications of convex programming, to prove facts about primal solutions, the main technique is to construct duals. In particular, to show that the SoS SDPs we use for exponential mechanism score functions satisfy bounded sensitivity and privacy, it suffices to construct SoS proofs witnessing these facts.

**Pseudoexpectations** For even  $d \in \mathbb{N}$ , a degree- $d$  pseudoexpectation in indeterminates  $x = x_1, \dots, x_n$  is a linear operator  $\tilde{\mathbf{E}} : \mathbb{R}[x_1, \dots, x_n]_{\leq d} \rightarrow \mathbb{R}$  which satisfies  $\tilde{\mathbf{E}} 1 = 1$  and  $\tilde{\mathbf{E}} p^2 \geq 0$  for every degree- $d/2$  polynomial  $p$ . We say that  $\tilde{\mathbf{E}}$  satisfies an inequality  $p(x) \geq 0$  if for every  $q$  such that  $\deg(p \cdot q^2) \leq d$  we have  $\tilde{\mathbf{E}} pq^2 \geq 0$ .

**Archimedean systems and duality** We say that a system of polynomial inequalities  $p_1(x) \geq 0, \dots, p_m(x) \geq 0$  is Archimedean if for some real  $M > 0$  it contains the inequality  $\|x\|^2 \leq M$ . SoS proofs and pseudoexpectations satisfy a natural duality for Archimedean systems, which we use often. Namely: for every Archimedean system  $p_1, \dots, p_m$  and every polynomial  $f$  and every degree  $d$ , exactly one of the following holds.

1. For every  $\varepsilon > 0$  there is an SoS proof  $p_1 \geq 0, \dots, p_m \geq 0 \vdash_d f \geq -\varepsilon$
2. There is a degree- $d$  pseudoexpectation satisfying  $p_1 \geq 0, \dots, p_m \geq 0$  but  $\tilde{\mathbf{E}} f < 0$ .

### 3.2 Privacy

We have already seen the definition of (pure) differential privacy. Our approach relies heavily upon the exponential mechanism of [MT07], we employ a volume-based version which appears in [KT13]. The proof is standard but we include it for completeness.

**Theorem 3.1** (volume-based exponential mechanism [MT07, KT13]). *The exponential mechanism  $M_E$  on inputs  $X, \mathcal{H} \subset \mathbb{R}^d$ ,  $s$ , selects and outputs some object  $h \in \mathcal{H}$ , where the probability a particular  $h$  is selected is proportional to  $\exp(\frac{\varepsilon s(X, h)}{2\Delta})$ . Let  $\mathcal{H}^* \subseteq \mathcal{H}$  be a set such that,  $\text{OPT}(X) \leq \inf_{h \in \mathcal{H}^*} s(X, h)$  be a lower bound for the score attained by the objects in  $\mathcal{H}^*$  with respect to the dataset  $X$ . Moreover, let  $\text{vol}(S)$  denote the Lebesgue measure of  $S$  in  $\mathbb{R}^d$ . Then*

$$\mathbb{P} \left[ s(M_E(X)) \leq \text{OPT}(X) - \frac{2\Delta}{\varepsilon} \left( \ln \left( \frac{\text{vol}(\mathcal{H})}{\text{vol}(\mathcal{H}^*)} + t \right) \right) \right] \leq \exp(-t).$$

*Proof.* We follow the same argument as the standard exponential mechanism analysis.

$$\begin{aligned} \mathbb{P}[s(M_E(X)) \leq c] &= \frac{\int_{h: s(X, h) \leq c, h \in \mathcal{H}} \exp\left(\frac{\varepsilon s(X, h)}{2\Delta}\right) dh}{\int_{h': h' \in \mathcal{H}} \exp\left(\frac{\varepsilon s(X, h')}{2\Delta}\right) dh'} \\ &\leq \frac{\text{vol}(\mathcal{H}) \exp\left(\frac{\varepsilon c}{2\Delta}\right)}{\text{vol}(\mathcal{H}^*) \exp\left(\frac{\varepsilon \text{OPT}(X)}{2\Delta}\right)} \\ &= \frac{\text{vol}(\mathcal{H})}{\text{vol}(\mathcal{H}^*)} \exp\left(\frac{\varepsilon(c - \text{OPT}(X))}{2\Delta}\right). \end{aligned}$$

From this inequality, the theorem statement can be obtained by substituting in the prescribed value for  $c$ . It remains to explain the first inequality. The numerator can be upper bounded since we are taking the integral at most over  $\mathcal{H}$ , and the value of the integral at each point is less than  $\exp(\varepsilon c/2\Delta)$ . Similarly, the denominator can be lower bounded by considering only the points in  $\mathcal{H}^*$ , all of which have a lower bound of  $\exp(\varepsilon \text{OPT}(X)/2\Delta)$ .  $\square$

We will also extensively use the following result of [BST14].

**Theorem 3.2** (Lemma 6.5 of [BST14]). *For every  $\varepsilon$  and  $d \in \mathbb{N}$ , there is an  $\varepsilon$ -DP algorithm  $A$  with the following guarantees. Given access to an evaluation oracle for a concave,  $L$ -Lipschitz function  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  and to membership and projection oracles for a convex set  $\mathcal{C} \subseteq \mathbb{R}^d$ , the algorithm produces a sample from a distribution  $D$  such that for every (measurable)  $S \subseteq \mathcal{C}$ ,*

$$e^{-\varepsilon} \mathbb{P}(A \in S) \leq \frac{\int_S \exp(f)}{\int_{\mathcal{C}} \exp(f)} \leq e^{\varepsilon} \mathbb{P}(A \in S).$$

*The algorithm runs in time  $\text{poly}(d, L \text{diam}(\mathcal{C}), 1/\varepsilon, \log \text{diam}(\mathcal{C}))$ , making at most that many queries to the oracles.*

[BST14] actually provides the running time bound  $\text{poly}(d, L, \text{diam}(\mathcal{C}), 1/\varepsilon)$ , but we can deduce the more precise bound in the theorem statement by a simple scaling argument. We note that recent work of [MV21] provides an algorithm which improves the running time's dependence on  $1/\varepsilon$  from polynomial to polylogarithmic, as well as reducing the polynomial dependence on  $d$  for convex sets  $\mathcal{C}$  that contain a ball of radius  $r$ , at an additional cost of  $\text{poly} \log(1/r)$  (Remark 2.5 of [MV21]). As our main focus is on providing polynomial time algorithms and not on designing the fastest algorithms, we do not further employ their improved methods.

## 4 Meta-Theorem on SoS Exponential Mechanism

To describe the SoS exponential mechanism more completely, we prove a meta-theorem on its performance. This meta-theorem could be extended in various ways, but the version we give here captures the mechanism as it is used in our paper. The reader interested solely in our result on mean estimation may comfortably skip this section as we do not rely on it elsewhere, but the exposition and abstraction here may make it easier to follow.

**Meta-theorem Setup** Consider the following template to capture the exponential mechanism in the language of polynomials. Let  $\mathcal{X}$  be a universe of possible datasets and  $\mathcal{C} \subseteq \mathbb{R}^n$  be a set of candidates. Consider a score function of the following form. For each dataset  $X$ , suppose there is a family of polynomials  $p^X, p_1^X, \dots, p_m^X$  in variables  $x_1, \dots, x_n, y_1, \dots, y_N$ , and the score function  $s(X, x)$  is given by

$$s(X, x) = \max_y p^X(x, y) \text{ such that } p_1^X(x, y) \geq 0, \dots, p_m^X(x, y) \geq 0.$$

**Example 4.1** (Tukey depth with margin). *To make things a bit less abstract, let us see that a score function closely related to the Tukey depth is expressible in this form. Suppose  $X = X_1, \dots, X_N \in \mathbb{R}^n$  is a dataset. Recall that the Tukey depth of a point  $z \in \mathbb{R}^n$  is given by the maximum number of  $X_i$ 's which lie on one side of a hyperplane through  $z$ .*

*Later on, for a fixed  $z \in \mathbb{R}^n$  with  $\|z\| = 1$ , we will want to run the exponential mechanism to select a direction  $v$  such that many  $X_i$ 's lie above a hyperplane through  $z$  in direction  $v$ , at distance at least distance  $r > 0$  from that hyperplane, so our score function for  $v$  is the number of such  $X_i$ 's. We can express this as follows:*

$$s(X, v) = \max_{b_1, \dots, b_N} \sum_{i=1}^N b_i \text{ s.t. for all } i, b_i^2 = b_i \text{ and } b_i \langle X_i - z, v \rangle \geq b_i \cdot r. \quad (2)$$

*Note that for fixed  $z$  and  $v$  it is easy to compute the number of  $X_i$ 's lying above the resulting hyperplane, but finding the maximizing  $v$  already appears to be a hard problem. (Of course, we want to accomplish only a related task: sampling from a distribution on high-scoring  $v$ 's.) Indeed, approximation algorithms for finding such  $v$  play a key role in recent algorithmic advances in robust and heavy-tailed statistics statistics.*

**Utility and Bounded Sensitivity in the Language of Polynomials** Utility in this framework takes exactly the same form as in the usual exponential mechanism – to demonstrate utility, one would show that high-scoring  $x$ 's are good, and there are not too many low-scoring  $x$ 's. The first of these statements is naturally expressible in the SoS proof system, and it turns out that the latter will not need to be expressed as an SoS proof. There is just one technical subtlety: the requirement that  $x \in \mathcal{C}$ , if it is used in the proof of utility, must be captured by the polynomials  $p_1(x, y) \geq 0, \dots, p_m(x, y) \geq 0$ , as the SoS proof system has no other way of natively using the hypothesis that  $x \in \mathcal{C}$ . (We illustrate this in the example below.)

To formulate bounded sensitivity within SoS will take a little additional work – SoS will require for there to be a certain kind of witness to bounded sensitivity. This is not a major restriction, as natural proofs of bounded sensitivity for optimization-based score functions typically yield such witnesses anyway.

Making this concrete, a natural way to show that a score function like the above satisfies bounded sensitivity is to relate feasible solutions to the optimization problem for  $X$  to those for

a neighboring dataset  $X'$ , without losing too much in the objective value. Let us suppose that for each neighboring pair  $X, X'$  there is a transformation  $y'(y)$  such that for all  $x$ , if  $y$  is feasible for  $X$  (i.e.  $p_1^X(x, y) \geq 0, \dots, p_m^X(x, y) \geq 0$ ) then  $y'(y)$  is feasible for  $X'$  (i.e.  $p_1^{X'}(x, y'(y)) \geq 0, \dots, p_m^{X'}(x, y'(y)) \geq 0$ ). If additionally  $p^X(x, y) - p^{X'}(x, y'(y)) \leq 1$ , this transformation (together with the corresponding one mapping  $y'$ s to  $ys$ ) witnesses bounded sensitivity for  $s$ . For technical reasons, our meta-theorem imposes the restriction that  $y'$  is a linear function of  $y$ , but this still suffices for our algorithms.

**Example 4.2** (Continuation of Example 4.1). *First addressing utility: it turns out that the proof of utility for  $v$  having high score according to (2) will rely on  $\|v\| \leq 1$ . So we will have to strengthen our system of polynomials to include this constraint. As a technicality, we will also shift our objective function so that having positive score is good enough for utility.*

$$\max_{b,v} \sum_{i=1}^N b_i - 0.9N \text{ such that } b_i^2 = b_i, b_i \langle X_i - z, v \rangle \geq b_i \cdot r, \|v\|^2 \leq 1. \quad (3)$$

Now, to establish bounded sensitivity, consider two neighboring datasets  $X = X_1, \dots, X_N$  and  $X' = X'_1, X_2, \dots, X_N$ , where  $X$  and  $X'$  differ on the first vector. If  $(b, v)$  is a feasible solution to (3) for  $X$  with objective value  $t$ , then we can replace it with  $(0, b_2, \dots, b_N, v)$  to get a feasible to solution for  $X'$  with objective value at least  $t - 1$ . The meta-theorem requires there to be an SoS proof of this fact; this (easy) SoS proof was first established in [Hop20].

**Robustly Satisfiable Polynomials** Before stating our meta-theorem, we need one more technical definition, capturing a certain well-conditioned-ness property of a polynomial optimization problem. Ultimately, this condition will imply a Lipschitz property of semidefinite relaxations of that optimization problem. This Lipschitz property will be used, in turn, to bound the running time of MCMC-based samplers for probability densities using those semidefinite relaxations as log-probabilities.

The details of the following definition (Definition 4.3) may appear opaque, but they are not too important – a good intuitive interpretation is that the polynomial optimization problem

$$\max_{x,y} p(x, y) \text{ s.t. } p_1(x, y) \geq 0, \dots, p_m(x, y) \geq 0$$

has a robust space of feasible solutions. Roughly, this means that for any  $x$  there is a small ball  $B$  around  $x$  such that for any  $x' \in B$  there is a feasible solution  $(x', y)$  whose objective value isn't too large. (The actual condition we use is slightly weaker than this.) This type of condition is common in meta-theorems involving the SoS proof system, to rule out the use of pathological  $p, p_1, \dots, p_m$ , and it is typically not too difficult to establish – see, e.g. [HKP<sup>+</sup>17, Wei17].

**Definition 4.3** (Robustly satisfiable polynomial systems). *Let  $\mathcal{C} \subseteq \mathbb{R}^n$  and let  $p, p_1, \dots, p_m$  be polynomials in  $x = x_1, \dots, x_n$  and  $y = y_1, \dots, y_N$ . Let  $\eta > 0$ . Consider a family of optimization problems, one for each  $x \in \mathcal{C}$ , given by*

$$\max_y p(x, y) \text{ s.t. } p_1(x, y) \geq 0, \dots, p_m(x, y) \geq 0.$$

*We say this family is  $\eta$ -robustly satisfiable if, for each  $x \in \mathcal{C}$ , each  $x'$  in the ball of radius  $\eta$  around  $x$  can satisfy the constraints. That is, for each  $x'$  such that  $\|x' - x\| \leq \eta$ , there exists  $y$  such that  $p_1(x', y) \geq 0, \dots, p_m(x', y) \geq 0$ .*



Note that in our algorithms,  $\eta$  will factor only into running times and not sample complexity or accuracy guarantees, so rather coarse bounds on  $\eta$ , perhaps loose by polynomial factors, suffice for our purposes.

**Example 4.4** (Continuation of Examples 4.1, 4.2). *Let us imagine now that  $\mathcal{C}$  is a ball of radius 0.9 centered at the origin, to see a proof sketch of  $\eta$ -robust satisfiability for (3). For each  $v \in \mathcal{C}$  and every vector  $\Delta$  with  $\|\Delta\| \leq 0.1$ , the constraints of (3) are satisfied by  $(v + \Delta, 0)$ . So, (3) is  $\eta$ -well-conditioned for  $\eta = 0.1$ , with respect to  $\mathcal{C}$ .*

With this setup in hand, we can state our meta-theorem.

**Theorem 4.5** (Meta-Theorem on SoS Exponential Mechanism). *Let  $\mathcal{C} \subseteq \mathbb{R}^n$  be a compact, convex set and  $\mathcal{X}$  a universe of possible datasets, equipped with a “neighbors” relation. Suppose that for every dataset  $X$  there exists an Archimedean and  $\eta$ -robustly satisfiable system of polynomial inequalities  $\mathcal{P}^X(x, y) = \{p_1^X(x, y) \geq 0, \dots, p_N^X(x, y) \geq 0\}$  and a polynomial  $p^X(x, y)$ , all of degree at most  $D$ , in indeterminates  $x_1, \dots, x_n, y_1, \dots, y_N$  such that for every neighboring dataset  $X'$  there is a linear function  $y'(y)$  such that bounded sensitivity has an SoS proof:*

$$\forall j, \mathcal{P}^X(x, y) \vdash_{\deg(p_j^{X'})} p_j^{X'}(x, y'(y)) \text{ and } \mathcal{P}^X(x, y) \vdash_D p^X(x, y) - p^{X'}(x, y') \leq 1.$$

*Suppose also that for every  $X$ , there are SoS proofs  $\mathcal{P}^X(x, y) \vdash_D p(x, y) \leq 1/\eta$  and  $\mathcal{P}^X(x, y) \vdash_D -p(x, y) \leq 1/\eta$ . Furthermore, suppose that the polynomials  $\mathcal{P}^X$  and  $p^X$ , and the polynomials used in the above SoS proofs, all have coefficients expressible in at most  $B$  bits.*

*Then for every  $\varepsilon > 0$  and  $D \in \mathbb{N}$  there exists an  $\varepsilon$ -differentially private algorithm which takes as input the polynomials  $p^X, p_1^X, \dots, p_m^X$  and  $B, \eta > 0$ , with the following guarantees:*

**Utility:** *For every  $X$ , if there is an SoS proof of utility for  $X$  which is degree- $\deg(p)$  with respect to  $p$ , i.e.,*

$$\mathcal{P}^X(x, y) \cup \{p(x, y) \geq 0\} \vdash_{D, \deg_p = \deg(p)} \|x - x^*(X)\|^2 \leq \alpha^2$$

*for some vector  $x^*(X) \in \mathbb{R}^n$  and  $\alpha > 0$ , where the coefficients of all polynomials involved in the proof are expressible with  $B$  bits, and if*

$$\frac{\text{vol}(\mathcal{C})}{\text{vol}(\{x \in \mathcal{C} : \exists y \text{ s.t. } \mathcal{P}^X(x, y) \text{ and } p^X(x, y) \geq t\})} \leq r,$$

*then the algorithm outputs  $x$  such that  $\|x - x^*(X)\| \leq \alpha + 2^{-B}$  with probability at least  $1 - r \exp(-\Omega(\varepsilon t))$ .*

**Running time:** *The algorithm runs in time*

$$\text{poly} \left( n^D, N^D, m^D, \frac{1}{\varepsilon}, \frac{1}{\eta}, \text{diam}(\mathcal{C}), B \right),$$

*making at most this many calls to membership and projection oracles for  $\mathcal{C}$ .*

## 4.1 Proof of Theorem 4.5

We describe the algorithm we use to prove Theorem 4.5; then we assemble the lemmas we need for the analysis.

**SoSExponentialMechanism** Input: polynomials  $p^X, p_1^X, \dots, p_m^X, D \in \mathbb{N}, \eta > 0, B \in \mathbb{N}$ .

1. For  $x_0 \in \mathcal{C}$ , let

$$s(X, x_0) = \max_{\tilde{\mathbf{E}}} \tilde{\mathbf{E}} p^X(x, y) \text{ such that } \deg \tilde{\mathbf{E}} = D, \tilde{\mathbf{E}} \text{ satisfies } \mathcal{P}^X, \text{ and } \tilde{\mathbf{E}} x = x_0$$

where the optimization is over  $\tilde{\mathbf{E}}$  in indeterminates  $x, y$ .

2. Let  $B' = B + T(\text{diam } \mathcal{C}, 1/\eta, 1/\varepsilon, d)$ , where  $T$  is a sufficiently-large polynomial in the running time of the log-concave private sampler of [BST14], when run with finite-precision arithmetic.
3. Run the log-concave private sampling algorithm of [BST14] (Lemma 6.5) with score function  $s$ , Lipschitz parameter  $\text{poly}(1/\eta)$ , and privacy parameter  $\varepsilon/4$ . Whenever the sampling algorithm makes a call to  $s(X, x_0)$ , solve the underlying SDP to  $\text{poly}(B')$  bits of precision.

Theorem 4.5 is immediate from the following lemmas, all of which we establish in the next section.

**Lemma 4.6** (High-scoring  $x_0$  is found in polynomial time). *Given the setup of Theorem 4.5, for all  $X$ , if*

$$\frac{\text{vol}(\mathcal{C})}{\text{vol}(\{x \in \mathcal{C} : \exists y \text{ s.t. } \mathcal{P}^X(x, y) \text{ and } p^X(x, y) \geq t\})} \leq r,$$

*then with probability at least  $1 - r \exp(-\varepsilon(t/2 - 1))$ , the  $x_0$  output in step (2) of SoSExponentialMechanism has  $s(X, x_0) \geq 0$ .*

**Lemma 4.7** (High-scoring  $x_0$  is useful). *Under the assumptions of Theorem 4.5, for all  $X$ , if there is an SoS proof of utility for  $X$  as described in Theorem 4.5, then for all  $x_0$  such that  $s(X, x_0) \geq 0$ ,  $\|x^*(X) - x_0\| \leq \alpha + 2^{-B}$ .*

**Lemma 4.8** (Privacy). *Under the assumptions of Theorem 4.5, SoSExponentialMechanism satisfies  $\varepsilon$ -DP.*

## 4.2 Proofs of Lemmas

We will prove Lemmas 4.6, 4.7, and 4.8.

**Remark on numerical issues:** Because of the choice of  $B'$ , the guarantees of the log-concave sampling algorithm will apply equally well if it receives  $s(X, x_0) \pm 2^{-B'}$  as if it receives  $s(X, x_0)$  when making oracle calls to  $s$ . (Given its running time, it cannot even read enough bits to tell the difference.) So, we will henceforth we will ignore the difference and presume that the log-concave sampler observes the values  $s(X, x_0)$  exactly.

The first step is to establish that the target probability distribution is actually log-concave and Lipschitz, so that we can use the guarantees of [BST14].

**Lemma 4.9.** *For all  $X$ , the function  $s(X, x_0)$  is concave in  $x_0$ .*

*Proof.* Consider  $x_0$  and  $x'_0$  and let  $\tilde{\mathbf{E}}$  be the optimal solution to the optimization problem defining  $s(X, x_0)$  and similarly for  $\tilde{\mathbf{E}}'$  and  $s(X, x'_0)$ . Then  $\frac{1}{2}\tilde{\mathbf{E}} + \frac{1}{2}\tilde{\mathbf{E}}'$  is feasible for  $S(X, \frac{1}{2}x_0 + \frac{1}{2}x'_0)$ . So  $s(X, \frac{1}{2}x_0 + \frac{1}{2}x'_0) \geq \frac{1}{2}s(X, x_0) + \frac{1}{2}s(X, x'_0)$ .  $\square$

**Lemma 4.10** (Robustly satisfiable systems yield Lipschitz SDPs). *Given the setup in Theorem 4.5, for all  $x_0, x'_0 \in \mathcal{C}$  and all  $X$ , we have  $|s(X, x_0) - s(X, x'_0)| \leq \text{poly}(1/\eta) \cdot \|x_0 - x'_0\|$ .*

*Proof.* As shorthand, let us write  $s = s(X, x_0)$ . Since  $\mathcal{P}^X$  is Archimedean, we can apply standard pseudoexpectation/SoS proof duality to conclude that for every  $\varepsilon > 0$  there is a polynomial identity in variables  $x, y$ :

$$s + \varepsilon - p^X(x, y) = \sum_{S \subseteq [m]} q_S(x, y) \prod_{i \in S} p_i^X(x, y) + \langle \lambda, x - x_0 \rangle,$$

where  $q_S$  are SoS polynomials, all the terms above have degree at most  $D$ , and  $\lambda \in \mathbb{R}^n$  is a vector.

Our first goal is to bound  $\|\lambda\|$ . By robust satisfiability, if we let  $x' = x_0 + \eta \cdot \frac{\lambda}{\|\lambda\|}$ , there exists  $y'$  such that  $p_i^X(x', y') \geq 0$  for all  $i$ . Hence,

$$s + \varepsilon - p^X(x', y') \geq \eta \|\lambda\|.$$

Since there are SoS proofs that  $p^X(x, y) \leq 1/\eta$  and  $-p^X(x, y) \leq 1/\eta$ , and we can take  $|\varepsilon| \leq 1/\eta$ , the left-hand side is  $O(1/\eta)$ , so we find  $\|\lambda\| \leq O(1/\eta^2)$ .

We claim that  $s(X, x'_0) \leq s + O(1/\eta^2) \cdot \|x_0 - x'_0\|$ . To see this, note that for each  $\varepsilon > 0$ , we can write

$$s + \varepsilon + \langle \lambda, x_0 - x'_0 \rangle - p^X(x, y) = \sum_{S \subseteq [m]} q_S(x, y) \prod_{i \in S} p_i^X(x, y) + \langle \lambda, x - x'_0 \rangle,$$

which, after Cauchy-Schwarz, certifies the upper bound  $s + \varepsilon + \|\lambda\| \|x_0 - x'_0\|$  on  $s(X, x'_0)$ . Since this works for all  $\varepsilon > 0$ , we find  $s(X, x'_0) \leq O(1/\eta^2) \|x_0 - x'_0\|$ .  $\square$

As a corollary of Lemmas 4.9 and 4.10, combined with Lemma 6.5 of [BST14], we obtain:

**Corollary 4.11.** *Given the setup of Theorem 4.5, for every  $X$ , the output of step (3) of SoSExponentialMechanism is a sample from a distribution  $D_X$  supported on  $\mathcal{C}$  such that for every event  $A$*

$$e^{-\varepsilon/2} \mathbb{P}_{D_X^{\text{target}}} (A) \leq \mathbb{P}_{D_X} (A) \leq e^{\varepsilon/2} \mathbb{P}_{D_X^{\text{target}}} (A)$$

where  $D_X^{\text{target}}$  is the distribution with density proportional to  $\exp((\varepsilon/2)s(X, x_0))$ . Furthermore, the sampler runs in time at most  $\text{poly}(d, \frac{1}{\varepsilon}, \frac{1}{\eta}, \text{diam}(\mathcal{C}))$ , making at most that many calls to a membership oracle for  $\mathcal{C}$  and to an evaluation oracle for  $s(X, \cdot)$ .

#### 4.2.1 Privacy: proof of Lemma 4.8

**Lemma 4.12.** *Given the conditions of Theorem 4.5, the score function  $s$  has sensitivity at most 1.*

*Proof.* Let  $X, X'$  be neighboring datasets, and let  $\tilde{\mathbf{E}}$  be an optimal solution to the optimization problem defining  $s(X, x_0)$ . We will construct a feasible solution  $\tilde{\mathbf{E}}'$  for  $s(X', x_0)$  whose objective value is at most  $s(X, x_0) + 1$ . Since we could swap  $X$  and  $X'$ , this will prove that  $|s(X, x_0) - s(X', x_0)| \leq 1$ .

For any degree  $D$  polynomial  $f$ , we define  $\tilde{\mathbf{E}}' f(x, y) = \tilde{\mathbf{E}} f(x, y'(y))$ . Note that the degree of  $\tilde{\mathbf{E}}'$  as a pseudoexpectation is the same as that of  $\tilde{\mathbf{E}}$ , because  $y'(y)$  is linear.

We claim that  $\tilde{\mathbf{E}}'$  is feasible for  $s(X', x_0)$ . Clearly  $\tilde{\mathbf{E}}' x = x_0$ , so we just need to check that  $\tilde{\mathbf{E}}'$  satisfies  $\mathcal{P}^{X'}$ . For each  $j$ , we check that  $\tilde{\mathbf{E}}'$  satisfies  $p_j^{X'}(x, y) \geq 0$ . Consider any square polynomial  $q$  such that  $\deg(q \cdot p_j^{X'}) \leq D$ . We need to show  $\tilde{\mathbf{E}}' q \cdot p_j^{X'} \geq 0$ .

Using the SoS proof  $\mathcal{P}^X(x, y) \vdash_{\deg(p_j^{X'})} p_j^{X'}(x, y'(y))$ , we can write

$$q(x, y') \cdot p_j^{X'}(x, y') = q(x, y') \cdot \sum_{S \subseteq [m]} q_S(x, y) \prod_{i \in S} p_i^X(x, y)$$

where every term in the sum on the right-hand side has degree at most  $\deg p_j^{X'}$ . Therefore, for every  $S$ , we have  $q(x, y'(y)) \cdot q_S(x, y) \prod_{i \in S} p_i^X(x, y)$  has degree at most  $D$ . So, applying  $\tilde{\mathbf{E}}'$  to both sides, we find that  $\tilde{\mathbf{E}}' q p_j^{X'} \geq 0$ , using that  $\tilde{\mathbf{E}}$  satisfies  $\mathcal{P}^X$ .

Finally, we have to check that  $\tilde{\mathbf{E}} p^X(x, y) - \tilde{\mathbf{E}}' p^{X'}(x, y) \leq 1$ . We expand the definitions and use our SoS proof of bounded sensitivity.

$$\tilde{\mathbf{E}} p^X(x, y) - \tilde{\mathbf{E}}' p^{X'}(x, y) = \tilde{\mathbf{E}} p^X(x, y) - \tilde{\mathbf{E}} p^{X'}(x, y') \leq 1.$$

□

*Proof of Lemma 4.8.* By Lemma 4.12 and the usual analysis of the exponential mechanism, an output from the target distribution  $D_X^{\text{target}}$  from Corollary 4.11 would satisfy  $\varepsilon/2$ -DP. Since, according to Corollary 4.11, the actual distribution output by step (3) of the algorithm differs only by multiplicative  $e^{\varepsilon/2}$ , the output of the log-concave sampler satisfies  $\varepsilon$ -DP. □

### 4.2.2 Utility: proofs of Lemmas 4.6 and 4.7

*Proof of Lemma 4.6.* By the standard analysis of the exponential mechanism, a sample from  $D_X^{\text{target}}$  (as described in Corollary 4.11) would output  $x_0$  with  $s(X, x_0)$  with probability at least  $1 - r \exp(-\varepsilon t/2)$ . Since the actual output distribution of step (3) is  $e^{\varepsilon/2}$  multiplicatively close to this, we are done. □

*Proof of Lemma 4.7.* Since  $x_0$  has  $s(X, x_0) \geq 0$ , there exists  $\tilde{\mathbf{E}}$  of degree  $D$  satisfying  $p_1^X(x, y) \geq 0, \dots, p_m^X(x, y) \geq 0$  and having  $\tilde{\mathbf{E}} x = x_0$  and  $\tilde{\mathbf{E}} p^X(x, y) \geq 2^{-\text{poly}(B)}$ . Then applying  $\tilde{\mathbf{E}}$  to either side of the SoS proof of utility, we obtain  $\tilde{\mathbf{E}} \|x - x^*(X)\|^2 \leq \alpha^2 + 2^{-\text{poly}(B)}$ . By convexity,  $\|\tilde{\mathbf{E}} x - x^*(X)\| \leq \alpha + 2^{-\text{poly}(B)}$ , so we are done. □

## 5 Coarse Estimation

Our overall algorithm for private mean estimation proceeds in two phases. At start of the algorithm, we assume that we have a point  $\mu_0$  such that  $\|\mu - \mu_0\|_2 \leq R$ . Equivalently, by re-centering, we assume that  $\|\mu\|_2 \leq R$ . Recall that packing lower bounds for differential privacy generally necessitate an assumption of this nature [HT10].

In the first phase of the algorithm we perform coarse mean estimation, improving our initial estimate of the mean from  $R$  to  $\mathcal{O}(\sqrt{d})$  error in  $\ell_2$ -norm. In the second phase, we perform fine mean estimation, further improving our estimate from  $\mathcal{O}(\sqrt{d})$  to  $\alpha$  error. The overall algorithm will follow by combining these two phases. In this section, we describe our algorithm for coarse mean estimation, quantified in Theorem 5.1.

**Theorem 5.1** (Combination of Theorem 5.5 and Theorem 5.10). *Suppose  $\mathcal{D}$  is a  $d$ -dimensional distribution with mean  $\mu$  and covariance  $\Sigma$ , such that  $\|\mu\|_2 \leq R$ ,  $\|\Sigma\|_2 \leq 1$ . Then there exists a polynomial-time  $\varepsilon$ -DP algorithm that takes  $n$  i.i.d. samples from  $\mathcal{D}$  and outputs  $y$  such that  $\|y - \mu\|_2 \leq \mathcal{O}(\sqrt{d})$  with probability  $1 - \beta$ , and sample complexity*

$$n = \tilde{\mathcal{O}} \left( \frac{d \log R + \min(d, \log R) \log(1/\beta)}{\varepsilon} \right).$$

To demystify the sample complexity in the above theorem, note that in order to prove this theorem, we present two algorithms with different sample complexities. The above sample complexity is the minimum of the two.

Our first algorithm uses a rather simple approach: privately estimate the mean in each coordinate up to constant error (using, e.g., the exponential mechanism, though one could also employ private histograms as in [KV18]), and use composition over the  $d$  coordinates. The sample complexity of this approach is  $\mathcal{O}(d(\log R + \log 1/\beta + \log d)/\varepsilon)$  (see Theorem 5.5). Similar methods have been previously employed for multivariate mean estimation (see, e.g., [KLSU19]), but we provide an analysis for completeness, and include a particular focus on the error probability  $\beta$ .

We observe that this coordinate-wise approach incurs a cost of  $d \log(1/\beta)$ , whereas the primary goal for heavy-tailed mean estimation in the non-private setting is to decouple these two factors. To this end, we give a second algorithm, which uses our SoS exponential mechanism paradigm. This algorithm has sample complexity  $\mathcal{O}(\log R(d + \log 1/\beta + \log \log R)/\varepsilon)$  (see Theorem 5.10). In the settings where  $\log R \ll d$ , this algorithm yields a better sample complexity than the simpler approach. An interesting open question is to design an efficient algorithm for this setting with sample complexity  $\tilde{\mathcal{O}}((d + \log R + \log(1/\beta))/\varepsilon)$ ; an appropriate application of the exponential mechanism provides an inefficient algorithm with this sample complexity.

## 5.1 First Algorithm: Coordinate-Wise

### 5.1.1 Main Steps

As the name suggests, our coordinate-wise algorithm works by running a one-dimensional algorithm on each coordinate of the samples from a  $d$ -dimensional distribution. The one-dimensional algorithm exploits the fact that the true mean is within constant distance of  $0.99n$  of the samples with high probability. We use the exponential mechanism to find any point that satisfies the same property. By triangle inequality, this new point will be close to the true mean.

In the following we describe the main theorem of this subsection and the main steps of proving it.

**Lemma 5.2.** *Suppose  $X_1, \dots, X_n$  are i.i.d. samples from a one-dimensional distribution  $\mathcal{D}$  with mean  $\mu$  and variance bounded above by 1. Furthermore, suppose  $n \geq 4 \log(1/\beta)/\alpha^2$ , and  $R^* \geq \sqrt{2/\alpha}$ . Then with probability  $1 - \beta$  there exists a set  $G \subseteq [n]$  such that  $|G| \geq (1 - \alpha)n$ , and*

$$\forall i \in G : |X_i - \mu| \leq R^* .$$

*Proof.* This lemma is a standard application of Chebyshev's inequality and Hoeffding's bound. Suppose the  $X_i$ 's are i.i.d. samples from  $\mathcal{D}$  and suppose  $\mathcal{D}$  has variance  $\sigma^2$ . Then by Chebyshev's inequality we know that

$$\mathbb{P}[|X_i - \mu| \geq R^*] \leq \frac{\sigma^2}{R^{*2}} ,$$

Let  $I_i = \mathbb{1}[|X_i - \mu| \geq R^*]$ , then  $I_i$ 's are independent random variables between 0 and 1. Therefore, we may use Hoeffding's bound. Let  $S_n = \sum_{i=1}^n I_i$ , then

$$\mathbb{P} \left[ S_n \geq \frac{\sigma^2}{R^{*2}} n + \sqrt{n \log 1/\beta} \right] \leq \beta .$$

Since  $R^* \geq \sqrt{2/\alpha}$ ,  $n \geq 4 \log(1/\beta)/\alpha^2$ ,  $\sigma^2 \leq 1$ , each of these terms are bounded by  $\alpha n/2$ , implying that  $|G| \geq (1 - \alpha)n$  as desired.  $\square$

We will prove the following theorem using a combination of the arguments from Section 5.1.2 to Section 5.1.5.

**Theorem 5.3** (COARSE-1D-ESTIMATE). *Let  $X_1, \dots, X_n \in \mathbb{R}$ ,  $R > R^* \geq 1$ ,  $0 \leq \alpha < 1/3$ . Suppose there exists a point  $x_0 \in [-R, R]$  such that  $|\{i \mid |X_i - x_0| \leq R^*\}| \geq (1 - \alpha)n$ . Then there exists an  $\varepsilon$ -DP algorithm and a universal constant  $C_0$  such that if  $n \geq C_0 (\log R + \log(1/\beta)) / \varepsilon \alpha$ , the algorithm returns  $y$  such that  $|y - x_0| \leq \mathcal{O}(R^*)$ , with probability  $(1 - \beta)$ , and runtime  $\mathcal{O}(n^2 \log R)$ .*

By combining Theorem 5.3 and Lemma 5.2, we arrive at the following theorem for univariate coarse mean estimation.

**Theorem 5.4** (one-dimensional coarse mean estimation). *Suppose  $\mathcal{D}$  is a one-dimensional distribution with mean  $\mu$ , and covariance  $\sigma^2$ , such that  $|\mu| \leq R$  and  $\sigma^2 \leq 1$ . Then there exists an  $\varepsilon$ -DP algorithm that takes  $n$  i.i.d. samples from  $\mathcal{D}$  and outputs  $y$  such that  $|y - \mu| \leq \mathcal{O}(1)$  with probability  $1 - \beta$ , sample complexity*

$$n = \mathcal{O}\left(\frac{\log R + \log(1/\beta)}{\varepsilon}\right),$$

and runtime  $\text{poly}(n, \log R, 1/\varepsilon)$ .

To extend this to the multivariate setting, we apply the algorithm from Theorem 5.4 with privacy budget  $\varepsilon/d$  and failure probability  $\beta/d$  to each coordinate ( $d$  times in total), and using composition of differential privacy.

**Theorem 5.5** (simple high dimensional coarse mean estimation). *Suppose  $\mathcal{D}$  is a  $d$ -dimensional distribution with mean  $\mu$  and covariance  $\Sigma$ , such that  $\|\mu\|_2 \leq R$  and  $\|\Sigma\|_2 \leq 1$ . Then there exists an  $\varepsilon$ -DP algorithm that takes  $n$  i.i.d. samples from  $\mathcal{D}$  and outputs  $y$  such that  $\|y - \mu\|_2 \leq \mathcal{O}(\sqrt{d})$  with probability  $1 - \beta$ , sample complexity*

$$n = \mathcal{O}\left(\frac{d \log R + d \log(1/\beta) + d \log d}{\varepsilon}\right),$$

and runtime  $\text{poly}(n, d, \log R, 1/\varepsilon)$ .

In the rest of this subsection we prove Theorem 5.3. To do so, we propose an algorithm and provide privacy, accuracy and runtime analysis for it.

### 5.1.2 Algorithm

Our algorithm (Algorithm 1) is based on the exponential mechanism. We construct a cover over  $[-R, R]$ , and run the exponential mechanism over it to capture a large number of  $X_i$ 's in a ball of radius  $\mathcal{O}(R^*)$ . This quantity is formalized using the score function as defined in Definition 5.6.

**Definition 5.6** (Score function). *Let  $X_1, \dots, X_n \in \mathbb{R}$ ,  $R^* \geq 1$ . For any point  $y \in \mathbb{R}$  we define*

$$\text{SCORE}(y) = |\{i \mid |X_i - y| \leq 2R^*\}|.$$

### 5.1.3 Privacy analysis

Since SCORE has sensitivity 1, the algorithm is  $\varepsilon$ -DP by privacy of the exponential mechanism.

---

**Algorithm 1** COARSE-1D-ESTIMATE

---

**Input:** Parameters  $R > R^* \geq 1$ , dataset  $X_1, \dots, X_n \in \mathbb{R}$ ,

**Operation:** 1. Let  $I = \{R^*k \mid \lfloor -R/R^* \rfloor \leq k \leq \lceil R/R^* \rceil, k \in \mathbb{Z}\}$ .

2. Run the exponential mechanism over the points in  $I$  with score function SCORE (Definition 5.6), privacy budget  $\varepsilon$ , and sensitivity 1, to obtain  $\hat{y}$ .

**Output:**  $\hat{y}$

---

### 5.1.4 Accuracy analysis

**Lemma 5.7.** *Suppose  $0 \leq \alpha < 1/3$ ,  $n \geq 2(\log(2\lceil R/R^* \rceil + 1) + \log(1/\beta))/\varepsilon\alpha$ , and that there exists some point  $x_0 \in [-R, R]$  such that  $|\{i \mid |X_i - x_0| \leq R^*\}| \geq (1 - \alpha)n$ . Then the exponential mechanism in Algorithm 1 will return  $\hat{y}$  such that*

$$|\hat{y} - x_0| \leq 3R^*,$$

with probability  $1 - \beta$ .

*Proof.* We show that there exists a point in  $I = \{R^*k \mid \lfloor -R/R^* \rfloor \leq k \leq \lceil R/R^* \rceil, k \in \mathbb{Z}\}$  that has SCORE at least  $(1 - \alpha)n$ . Let  $G$  be the set of points which are close to the point  $x_0$ , that is,  $G = \{i \mid |X_i - x_0| \leq R^*\}$  and  $|G| \geq (1 - \alpha)n$ . Let  $y^*$  be the point in  $I$  that is closest to  $x_0$ . Since  $x_0 \in [-R, R]$ ,  $|x_0 - y^*| \leq R^*$ . Then

$$\forall i \in G : |y^* - X_i| \leq |y^* - x_0| + |x_0 - X_i| \leq R^* + R^* \leq 2R^*.$$

Therefore  $y^*$  has SCORE at least  $(1 - \alpha)n$ . By the guarantees of the standard exponential mechanism, the exponential mechanism when instantiated with the SCORE function over  $I$  will return  $\hat{y}$  such that

$$\text{SCORE}(\hat{y}) \geq (1 - \alpha)n - \frac{2}{\varepsilon}(\log(2\lceil R/R^* \rceil + 1) + \log(1/\beta)),$$

with probability  $1 - \beta$ . Since  $\alpha n \geq \frac{2}{\varepsilon}(\log(2\lceil R/R^* \rceil + 1) + \log(1/\beta))$ , the exponential mechanism will return  $\hat{y}$  such that

$$\text{SCORE}(\hat{y}) \geq (1 - 2\alpha)n,$$

with probability  $1 - \beta$ . Let  $T = \{i \mid |X_i - \hat{y}| \leq 2R^*\}$ , then  $|T| \geq (1 - 2\alpha)n$ , with probability  $1 - \beta$ . Moreover  $G = \{i \mid |X_i - x_0| \leq 2R^*\}$  has at least  $(1 - \alpha)n$  members. Therefore since  $\alpha < 1/3$ , we have that  $T \cap G \neq \emptyset$ , and therefore there exists some  $j \in T \cap G$ . Hence  $|X_j - x_0| \leq R^*$  and  $|X_j - \hat{y}| \leq 2R^*$ . Therefore  $|y - x_0| \leq 3R^*$  with probability  $1 - \beta$ , as desired.  $\square$

### 5.1.5 Runtime analysis: lazy exponential mechanism

If the exponential mechanism in Algorithm 1 is naively implemented it will take  $\Theta(Rn)$  time. This is a pseudo-polynomial time algorithm, but we would prefer a truly polynomial time algorithm, with running time polynomial in  $\log R$ . In the following theorem, we describe how this application of the exponential mechanism can be implemented in  $\mathcal{O}(n^2 \log R)$  time. This style of “lazy exponential mechanism” may be folklore, but we include the argument for completeness.

**Theorem 5.8** (lazy exponential mechanism). *Suppose  $C \in \mathbb{N}$ ,  $I = \{b_1, \dots, b_m\}$  is a set of bins, and  $n \in \mathbb{N}$  objects  $O = \{o_1, \dots, o_n\}$  are given. Moreover suppose that each item can belong to at most  $C$  bins, and that for each object  $o$ , the set of the bins it belongs to can be computed in time  $\mathcal{O}(C \log m)$ . For a bin  $b_i$ , let*

$$\text{SCORE}(b_i) = |\{j \mid o_j \text{ belongs to } b_i\}|.$$

*Then the exponential mechanism over  $I$  with score function SCORE can be implemented in time  $\mathcal{O}(Cn^2 \log m)$ .*

*Proof.* Let  $J = \{b_i \mid \text{SCORE}(b_i) > 0\}$ . We can construct  $J$  and compute the score function for all of its members by iterating over  $O$  and finding which bins each of them belongs to. This will take time  $\mathcal{O}(Cn \log m)$ . Now since all of the other bins are empty, the probability that the exponential mechanism with sensitivity  $\Delta$  and privacy budget  $\varepsilon$  selects  $b_j \in J$  has to be

$$\mathbb{P}[b_j] := \frac{\exp\left(\frac{\varepsilon \text{SCORE}(b_j)}{2\Delta}\right)}{\sum_{k, b_k \in J} \exp\left(\frac{\varepsilon \text{SCORE}(b_k)}{2\Delta}\right) + m - |J|}.$$

Therefore the probability of choosing each member of  $J$  can be computed in time  $\mathcal{O}(n \log m)$ . Moreover, let  $\perp$  be a special symbol representing the event that the exponential mechanism selects a  $b_j$  such that  $b_j \notin J$ . It is easy to see that

$$\mathbb{P}[\perp] := \frac{m - |J|}{\sum_{k, b_k \in J} \exp\left(\frac{\varepsilon \text{SCORE}(b_k)}{2\Delta}\right) + m - |J|}.$$

We now draw an element from the set  $\{b_j \mid j \in J\} \cup \{\perp\}$  where the probabilities are as established above. If the sampling algorithm picks a  $b_j \in J$  we can return that, and we are done. Otherwise, if  $\perp$  is picked, we can sample an index from  $i \in \{1, \dots, m - |J|\}$  uniformly at random in time  $\mathcal{O}(\log m)$ . Now we aim to construct a uniformly random member from  $I - J$  from this random index  $i$ . To do we just need to consider  $K = \{b_i, \dots, b_{i+|J|}\}$ . By the intermediate value theorem, there exists a member  $b_{i+k} \in K$  such that  $k$  equals the number of the members of  $J$  which have index smaller than or equal to  $i + k$ . For the smallest  $k$  that has such property, we return  $b_{i+k}$ , it is easy to see that this scheme returns a uniformly random member from  $I - J$ . Moreover, the overall algorithm takes time at most  $\mathcal{O}(Cn^2 \log m)$ . Note that in this proof, the  $\log m$  factors come from the bit complexity of indices of the members of  $I$ .  $\square$

**Corollary 5.9.** *The exponential mechanism in Algorithm 1 can be implemented in time  $\mathcal{O}(n^2 \log R)$ .*

*Proof.* Each  $X_i$  can be counted in the SCORE of at most 5 members of  $I$ . Furthermore, for each  $X_j$ , to find the members of  $I$  for which it is counted in, we just need to find the points in  $[X_i - 2R^*, X_i + 2R^*]$  which are of the form  $kR^*$ , for some  $k \in \mathbb{Z}$ . This can be done in time  $\mathcal{O}(\log R)$ . The statement thus holds by an application of Theorem 5.8.  $\square$

## 5.2 Second Algorithm: SoS Exponential Mechanism

### 5.2.1 Main Steps

We proceed to describe our second coarse-estimation algorithm, based on our SoS Exponential Mechanism framework. Our coverage is for a concrete problem (as opposed to the abstract setting in our meta-theorem in Section 4) and conceptually simpler than our fine-estimation algorithm (Section 6), so it may be a good starting point for a reader to understand the framework.



Consider the following simple but inefficient method for performing coarse estimation. Let the score function for a point  $v$  be the number of datapoints within a ball of radius  $\text{poly}(d)$  centered at  $v$ . One could run the exponential mechanism with this score function, with the set of candidate points being the ball of radius  $R$ . Compared to the coordinate-wise algorithm described in Section 5.1, this approach works directly in the high-dimensional space.

Of course, this algorithm is not polynomial time, as the naive method of implementing it would involve computing the score function for all points in a cover of size exponential in  $d$ . Instead, appealing to our SoS Exponential mechanism framework, we first replace the score function with a similar SoS optimization problem. The resulting algorithm and optimization problem are both described in Section 5.2.2. Note that for technical reasons, we actually run the exponential mechanism  $\mathcal{O}(\log R)$  times, making a constant factor improvement in the radius each time. This iterative process is what gives rise to a sub-optimal  $\mathcal{O}(\log R \log(1/\beta)/\varepsilon)$  term in the sample complexity. To prove accuracy of the overall procedure, we first show that the solution to the SoS optimization problem will give a point with high utility (Lemma 5.14 in Section 5.2.3). Then, assuming the instantiations of the exponential mechanism are all “successful” (that is, produce points which are of comparable utility to the one which optimizes the score function), it is not hard to show that the final point output will satisfy our desired accuracy guarantee (Corollary 5.16 in Section 5.2.4).

At this point, there are three components of the exponential mechanism left to argue: privacy, accuracy, and computational efficiency. Privacy is fairly straightforward: we prove that the score function has bounded sensitivity (Lemma 5.17 in Section 5.2.5), which implies privacy (Corollary 5.18) by combining the exponential mechanism’s privacy guarantee with basic composition. Utility follows the standard recipe for the volume-based exponential mechanism (Theorem 3.1): in addition to the aforementioned sensitivity bound, we require that points with high utility are sufficiently dense (Lemma 5.21 in Section 5.2.7). Combining the two gives the desired accuracy guarantee in Lemma 5.23 in Section 5.2.9. Finally, to ensure that the algorithm is polynomial time, we employ the sampling-based exponential mechanism of [BST14] (Theorem 3.2). We again use the fact that the score function has bounded sensitivity, but also Lipschitzness (Lemma 5.20 in Section 5.2.6), and concavity (Corollary 5.22 in Section 5.2.8). Combining these properties allow us to conclude the exponential mechanism can be implemented in polynomial time (Lemma 5.24 in Section 5.2.10).

These allow us to conclude the following result for coarse mean estimation.

**Theorem 5.10** (high dimensional coarse mean estimation via SoS exponential mechanism). *Suppose  $\mathcal{D}$  is a  $d$ -dimensional distribution with mean  $\mu$  and covariance  $\Sigma$ , where  $\|\mu\|_2 \leq R$ , and  $\|\Sigma\|_2 \leq 1$ . Then there exists a polynomial time  $\varepsilon$ -DP algorithm COARSE-ESTIMATE that takes*

$$n = \mathcal{O}\left(\frac{\log R (d + \log(1/\beta) + \log \log R)}{\varepsilon}\right)$$

*i.i.d. samples from  $\mathcal{D}$  and outputs  $y$ , such that  $\|y - \mu\|_2 \leq \mathcal{O}(\sqrt{d})$  with probability  $1 - \beta$ .*

Before we begin the main analysis, we establish a deterministic condition on the dataset under which our algorithm succeeds. This condition holds with high probability if the dataset is sampled from an appropriate distribution. Specifically, the following is a multivariate analogue of Lemma 5.2, which states that most samples from a distribution with bounded covariance will be within a bounded distance of the mean. The proof is similar to before.

**Lemma 5.11.** *Suppose  $X_1, \dots, X_n$  are i.i.d. samples from a  $d$ -dimensional distribution  $\mathcal{D}$ , with mean  $\mu$  and covariance  $\Sigma$ , where  $\|\Sigma\|_2 \leq 1$ . Furthermore suppose  $n \geq 4 \log(1/\beta)/\alpha^2$  and  $R^* \geq \sqrt{2d/\alpha}$ . Then with probability  $1 - \beta$ , there exists a set  $G$  such that  $|G| \geq (1 - \alpha)n$ , and*

$$\forall i \in T : \|X_i - \mu\|_2 \leq R^* .$$

With this in place, it suffices to prove the following theorem, which is our main technical result of this section.

**Theorem 5.12.** *For every  $\varepsilon > 0$  and  $n, d \in \mathbb{N}$  there is an  $\varepsilon$ -differentially private polynomial-time algorithm  $\text{COARSE-ESTIMATE}(R, R^*, X_1, \dots, X_n)$  with the following guarantees. There exists a universal constant  $C_0$  that given  $X_1, \dots, X_n \in \mathbb{R}^d$ ,  $R, R^* > 0$ ,  $\text{COARSE-ESTIMATE}$  outputs  $y \in \mathbb{R}^d$  such that if there exists  $x \in \mathbb{R}^d$  with  $\|x\|_2 \leq R$  and such that  $|\{X_i : \|X_i - x\|_2 \leq R^*\}| \geq (1 - \alpha)n$ ,  $R \geq 1000R^* > 1$ ,  $n \geq C_0 \frac{\log R}{\alpha \varepsilon} (d + \log(1/\beta) + \log \log R)$ ,  $0 < \alpha < 10^{-4}$ , then  $\|y - x\|_2 \leq O(R^*)$ , with probability at least  $1 - \beta$ .*

In the rest of this section (from Section 5.2.2 to Section 5.2.10) we prove Theorem 5.12. To do so, we present an algorithm and provide privacy, accuracy, and runtime analysis for it.

### 5.2.2 Algorithm

We describe the score function employed in the exponential mechanism. Our algorithm repeatedly invokes the exponential mechanism with a geometrically decreasing radius.

**Definition 5.13** (COARSE-SDP). *Let  $X_1, \dots, X_n, y \in \mathbb{R}^d$ , and  $R > r > 1$ . Let COARSE-SDP be the following convex program, where the maximization is over degree-4 pseudo-distributions in indeterminates  $v_1, \dots, v_d, b_1, \dots, b_n$ .*

$$\begin{aligned} \max_{\tilde{\mathbf{E}}} \quad & \tilde{\mathbf{E}} \sum_{i=1}^n b_i \text{ s.t.} \\ & \tilde{\mathbf{E}} \text{ satisfies } b_i^2 = b_i \quad \forall i \\ & \tilde{\mathbf{E}} \text{ satisfies } \|v\|_2^2 \leq (2R + R/1000)^2 \\ & \tilde{\mathbf{E}} \text{ satisfies } b_i \|X_i - v\|_2^2 \leq b_i \cdot r^2 \quad \forall i \\ & \tilde{\mathbf{E}} v = y. \end{aligned}$$

---

#### Algorithm 2 COARSE-ESTIMATE

---

**Input:** Parameters  $R \geq 1000R^* \geq 1$ , dataset  $X_1, \dots, X_n \in \mathbb{R}^d$

**Operation:** Let  $m = \lfloor \log_5(R/1000R^*) \rfloor$ . For  $0 \leq t < m$  do:

1. Let  $R_t = R(0.2)^t$ .
2. Run the exponential mechanism with score function COARSE-SDP ( $R \leftarrow R_t, r \leftarrow R_t/100$ ), privacy budget  $\varepsilon/m$ , and sensitivity 1 over the ball of radius  $R_t + R_t/1000$  in  $\mathbb{R}^d$  to obtain  $y_t$ .
3. For every  $j$  let  $X_j = X_j - y_t$ .

**Output:**  $\hat{y} = \sum_{i=0}^{m-1} y_i$

---

### 5.2.3 SoS Analysis

We prove that a point with a high score function value will also have high utility by containing a large fraction of the dataset within a bounded radius.

**Lemma 5.14.** Let  $X_1, \dots, X_n \in \mathbb{R}^d$ ,  $R > r > R^* > 1$ , and  $\alpha \in [0, 1/3]$  and let  $\tilde{\mathbf{E}}$  be a feasible solution to COARSE-SDP with  $\tilde{\mathbf{E}} \sum_{i \leq n} b_i \geq (1 - 2\alpha)n$ . For any  $x_0 \in \mathbb{R}^d$  such that  $|\{i : \|X_i - x_0\|_2 \leq R^*\}| \geq (1 - \alpha)n$  and  $\|x_0\|_2 \leq R$ ,

$$\|y - x_0\|_2 < 10(r + \sqrt{\alpha}R).$$

*Proof.* Let  $G = \{i : \|X_i - x_0\| \leq R^*\}$  be the *good* set of indices. Since  $\sum_{i \in G} \tilde{\mathbf{E}}[b_i] + \sum_{i \notin G} \tilde{\mathbf{E}}[b_i] = \sum_{i=1}^n \tilde{\mathbf{E}}[b_i] \geq (1 - 2\alpha)n$ , and  $|G| \geq (1 - \alpha)n$ ,

$$\sum_{i \in G} \tilde{\mathbf{E}}[b_i] \geq (1 - 2\alpha)n - (n - |G|) \geq (1 - 3\alpha)n.$$

We have

$$|G| \cdot \|x_0 - v\|^2 = \sum_{i \in G} b_i \cdot \|x_0 - v\|^2 + \sum_{i \notin G} (1 - b_i) \cdot \|x_0 - v\|^2.$$

We will bound the two terms separately. For the first, for any  $i \in G$ , by SoS triangle inequality (Fact B.7) we have

$$\tilde{\mathbf{E}} b_i \|x_0 - v\|^2 \leq 2 \tilde{\mathbf{E}} b_i (\|x_0 - X_i\|^2 + \|X_i - v\|^2) \leq 4r^2 \cdot \tilde{\mathbf{E}} b_i.$$

(Here we used that  $\tilde{\mathbf{E}}$  satisfies  $b_i \geq 0$ .) For the second term, again using triangle inequality, we have

$$\tilde{\mathbf{E}}(1 - b_i) \|x_0 - v\|^2 \leq 2 \tilde{\mathbf{E}}(1 - b_i) (\|v\|_2^2 + \|x_0\|_2^2) \leq 2 \tilde{\mathbf{E}}(1 - b_i) (R^2 + (2R + R/1000)^2) \leq 11R^2 \tilde{\mathbf{E}}(1 - b_i).$$

(Here we used that  $\tilde{\mathbf{E}}$  satisfies  $1 - b_i \geq 0$ .) So overall,

$$|G| \cdot \tilde{\mathbf{E}} \|x_0 - v\|^2 \leq 4r^2 \cdot \tilde{\mathbf{E}} \sum_{i \in G} b_i + 11R^2 \cdot \tilde{\mathbf{E}} \sum_{i \in G} (1 - b_i).$$

Now  $\sum_{i \in G} \tilde{\mathbf{E}} b_i \geq (1 - 3\alpha)n$ , so  $\tilde{\mathbf{E}} \sum_{i \in G} (1 - b_i) \leq 3\alpha n$ , and hence

$$\tilde{\mathbf{E}} \|x_0 - v\|^2 \leq 11 \cdot \frac{nr^2 + 3\alpha n R^2}{|G|} \leq 66(r^2 + \alpha R^2) < 100(r^2 + \alpha R^2).$$

Since  $\|\tilde{\mathbf{E}} v - x_0\| \leq \sqrt{\tilde{\mathbf{E}} \|v - x_0\|^2}$ , we are done.  $\square$

## 5.2.4 Accuracy Analysis Assuming Exponential Mechanism's Success

Here we show that if each iteration of Algorithm 2 produces a high utility solution, the resulting estimates will improve in accuracy at a rate exponential in the number of iterations.

**Proposition 5.15.** Suppose  $|\{i : \|X_i - x_0\|_2 \leq R^*\}| \geq (1 - \alpha)n$ ,  $\alpha \leq 10^{-4}$ ,  $\|x_0\|_2 \leq R$ , and assume the exponential mechanism in Algorithm 2 is successful in every step, i.e.,  $y_t$  has score larger than  $(1 - 2\alpha)n$  for every  $t$ . Then for each  $t$ ,

$$\left\| \sum_{i=0}^{t-1} y_i - x_0 \right\|_2 \leq R_t.$$

*Proof.* We prove this proposition through induction. For the base case  $t = 0$ , the sum  $\sum_{i=0}^{t-1} y_i$  is empty and we are left with  $\|x_0\| \leq R_0 = R$ , which holds by assumption. For the inductive case, we suppose the statement holds for  $t - 1$ , and prove that it holds for  $t$ . The inductive hypothesis is that  $\|\sum_{i=0}^{t-2} y_i - x_0\|_2 \leq R_{t-1}$ . We may now use Lemma 5.14, where  $x_0$  and  $R$  in the lemma statement are replaced by  $\sum_{i=0}^{t-2} y_i - x_0$  and  $R_{t-1}$ , respectively. This gives us

$$\left\| \sum_{i=0}^{t-1} y_i - x_0 \right\|_2 \leq 10(R_{t-1}/100 + R_{t-1}/100) \leq R_{t-1}/5 = R_t,$$

as desired.  $\square$

As an immediate corollary, we can bound the resulting error after Algorithm 2 is complete.

**Corollary 5.16.** *Under the same assumptions as Proposition 5.15 and noting that  $m = \log_5(R/1000R^*)$ , Algorithm 2 outputs  $y$ , such that*

$$\|y - x_0\|_2 \leq 1000R^*.$$

### 5.2.5 Sensitivity

We now show that the score function has bounded sensitivity.

**Lemma 5.17** (COARSE-SDP sensitivity). *COARSE-SDP has sensitivity 1 with respect to the  $X_i$ 's.*

*Proof.* Without loss of generality suppose  $X, X'$  are two datasets that differ in the first index. Suppose for inputs  $(X_1, X_2, \dots, X_n)$ ,  $\tilde{\mathbf{E}}$  is the degree-4 pseudo-expectation that maximizes the objective function of COARSE-SDP. In order to prove that COARSE-SDP has sensitivity 1, we prove that there exists a feasible solution  $\tilde{\mathbf{E}}'$  to the COARSE-SDP convex program, on inputs  $(X'_1, X_2, \dots, X_n)$ , such that

$$\tilde{\mathbf{E}}'[\sum b_i] \geq \tilde{\mathbf{E}}[\sum b_i] - 1.$$

If we do so, sensitivity follows by symmetry. To construct  $\tilde{\mathbf{E}}'$ , we describe it by the pseudo-moments up to the fourth degree. For every degree-4 monomial  $f$  in  $b, v$  that does not contain  $b_1$ , let

$$\tilde{\mathbf{E}}'[f(b, v)] = \tilde{\mathbf{E}}[f(b, v)],$$

and for every degree-4 monomial  $f$  in  $b, v$ , that contains  $b_1$ , let

$$\tilde{\mathbf{E}}'[f(b, v)] = 0.$$

Then  $\tilde{\mathbf{E}}'$  satisfies the constraints in COARSE-SDP. Furthermore,  $\tilde{\mathbf{E}}'[\sum b_i] \geq \tilde{\mathbf{E}}[\sum b_i] - 1$ . It remains to show that  $\tilde{\mathbf{E}}'$  is a pseudo-expectation. This is because  $\tilde{\mathbf{E}}'[(1, b, v)^{\otimes 2}((1, b, v)^{\otimes 2})^T]$  is PSD, which is a direct result of Lemma B.1.  $\square$

Combining this with the privacy guarantees of the exponential mechanism and basic composition, we see that the overall algorithm is  $\epsilon$ -DP.

**Corollary 5.18** (privacy analysis). *The exponential mechanism in Algorithm 2 is  $\epsilon/m$ -DP. Hence Algorithm 2 is  $\epsilon$ -DP by composition.*

### 5.2.6 Lipschitzness

To prove that the score function is Lipschitz, we rely upon the following lemma, which is a direct result of the duality of pseudo-distributions and SoS proofs (see, e.g., [BS16]).

**Lemma 5.19.** *Suppose  $R, r, X_1, \dots, X_n$  are fixed. Let  $f(y)$  denote the maximum value COARSE-SDP can obtain for  $y$ . Then for every  $y \in \mathbb{R}^d, \omega > 0$ , there exists  $\alpha_i \in \mathbb{R}[b, v]_{\leq 2}$ , a degree-2 SoS polynomial  $\beta, \gamma_i \geq 0, \lambda_i \in \mathbb{R}$ , and degree-4 SoS polynomial  $\sigma$  such that*

$$\begin{aligned} f(y) + \omega - \sum_{i=1}^n b_i &= \sum_{i=1}^n \alpha_i(b, v)(b_i^2 - b_i) + \beta(b, v)((2R + R/1000)^2 - \|v\|_2^2) \\ &\quad + \sum_{i=1}^n \gamma_i b_i (r^2 - \|X_i - v\|_2^2) + \sum_{i=1}^d \lambda_i (v_i - y_i) + \sigma(b, v). \end{aligned}$$

Note that the above equality is a polynomial identity in  $b$  and  $v$ .

Note that the constraints on  $b$  and  $v$  in the optimization problem imply boundedness (Lemma A.3) and therefore our problem satisfies the duality conditions. We use this lemma to prove the following theorem.

**Lemma 5.20** (COARSE-SDP lipschitzness). *In the convex optimization problem COARSE-SDP, suppose  $R, r, X_1, \dots, X_n$  are fixed. Let  $f(y)$  denote the maximum value COARSE-SDP can obtain for  $y$ . Then  $f(y)$  is  $L$ -Lipschitz over the ball of radius  $R + R/1000$  in  $\mathbb{R}^d$ , where  $L = n\sqrt{d}/R$ .*

*Proof.* Let

$$\begin{aligned} g(\alpha, \beta, \gamma, \sigma; y) &= \sum_{i=1}^n b_i + \sum_{i=1}^n \alpha_i(b, v)(b_i^2 - b_i) + \beta(b, v)((2R + R/1000)^2 - \|v\|_2^2) \\ &\quad + \sum_{i=1}^n \gamma_i b_i (r^2 - \|X_i - v\|_2^2) + \sum_{i=1}^d \lambda_i (v_i - y_i) + \sigma(b, v). \end{aligned}$$

Note that  $g$  is a polynomial in  $b$  and  $v$ . Then by Lemma 5.19 we know that for any  $y \in \mathbb{R}^d, \omega > 0$ , there exists  $\alpha, \beta, \gamma, \lambda, \sigma$  such that

$$f(y) + \omega = g(\alpha, \beta, \gamma, \lambda, \sigma; y), \quad (4)$$

is a polynomial identity in  $b$  and  $v$ . First, we prove that the  $\lambda_i$ 's are bounded. To do so we plug different values for  $b$  and  $v$  in Equation (4). Let  $b = 0$ , and  $v = y + te_j$ , such that  $\|y + te_j\|_2 = 2R + R/1000$ , and  $t > 0$ . Such  $t$  exists and is larger than  $R$  due to  $\|y\|_2 \leq R + R/1000$ , and the triangle inequality. Plugging this  $b$  and  $v$  into Equation (4), gives us

$$f(y) + \omega \geq \lambda_j t.$$

Since  $t \geq R$ ,  $f(y) + \omega \geq R\lambda_j$ . Similarly, by plugging  $b = 0, v = y - te_j$ , we obtain  $f(y) + \omega \geq -R\lambda_j$ , and therefore,  $|\lambda_j| \leq (f(y) + \omega)/R$ . Since  $f(y) \leq n$ , we obtain  $|\lambda_i| \leq (n + \omega)/R$ , and hence

$$\|\lambda\|_2 \leq \sqrt{d}(n + \omega)/R.$$

Now that we have bounded  $\lambda$ , we can prove that  $f$  is Lipschitz. To do so, suppose  $y' = y + \delta u$ , where  $\|u\|_2 = 1$ . If for  $y'$  we can construct a dual solution that obtains value  $f(y) + L\delta + h(\omega)$ ,

where  $\lim_{\omega \rightarrow 0} h(\omega) = 0$ , we have proven that  $f(y') \leq f(y) + L\delta$ , and Lipschitzness will follow by symmetry. To construct this dual solution, let  $(\alpha', \beta', \gamma', \lambda', \sigma') = (\alpha, \beta, \gamma, \lambda, \sigma)$ . Then

$$\begin{aligned} g(\alpha', \beta', \gamma', \lambda', \sigma'; y') &= g(\alpha, \beta, \gamma, \lambda, \sigma; y) + \delta \langle \lambda, u \rangle \\ &\leq f(y) + \omega + \delta \|\lambda\|_2 \\ &\leq f(y) + \omega + \delta \sqrt{d}(n + \omega)/R \\ &\leq f(y) + \delta \sqrt{d}n/R + \omega(1 + \delta \sqrt{d}/R), \end{aligned}$$

as desired. Therefore  $f$  is  $L = n\sqrt{d}/R$ -Lipschitz.  $\square$

### 5.2.7 Volume

We show that the set of points with high score function is sufficiently dense. This essentially follows from the fact that every point which is close to the optimal solution also has high utility, and by an argument about the volume of  $\ell_2$ -balls.

**Lemma 5.21** (COARSE-SDP volume). *Let  $R > 1000R^* > 1$ . Suppose there exists a point  $x_0 \in \mathbb{R}^d$  such that  $|\{i : \|X_i - x_0\|_2 \leq R^*\}| \geq (1 - \alpha)n$  and  $\|x_0\|_2 \leq R$ . Let  $\mathbb{B}$  denote the ball of radius  $R + R/1000$  in  $\mathbb{R}^d$ . Then there exists a set  $\mathcal{H}^* \subseteq \mathbb{B}$ , such that for every point  $y$  in  $\mathcal{H}^*$ , the value of COARSE-SDP with  $(R \leftarrow R, r \leftarrow R/100)$  is larger than  $(1 - \alpha)n$  and*

$$\log \frac{\text{vol}(\mathbb{B})}{\text{vol}(\mathcal{H}^*)} \leq d \log(2000).$$

*Proof.* Let  $\mathcal{H}^*$  be the ball of radius  $R/1000$  centered at  $x_0$ . Then  $\mathcal{H}^* \subseteq \mathbb{B}$ , and

$$\log \frac{\text{vol} \mathbb{B}}{\text{vol}(\mathcal{H}^*)} = \log \left( \frac{R + R/1000}{R/1000} \right)^d \leq d \log(2000).$$

It remains to prove that every point in  $\mathcal{H}^*$  has high COARSE-SDP value. Suppose  $y$  is a point in  $\mathcal{H}^*$ . We know that COARSE-SDP is a relaxation of the polynomial optimization problem; therefore, if we propose a solution to the polynomial optimization problem with value  $(1 - \alpha)n$ , we have proven that COARSE-SDP has value at least  $(1 - \alpha)n$ . Hence, it remains to show that there exists a set of size at least  $(1 - \alpha)n$  from the  $X_i$ 's such that  $\|X_i - y\|_2 \leq R/100$ . Consider the set  $G = \{i : \|X_i - x_0\|_2 \leq R^*\}$ . We know that  $|G| \geq (1 - \alpha)n$ . Moreover,

$$\forall i \in G : \|X_i - y\|_2 \leq \|X_i - x_0\|_2 + \|x_0 - y\|_2 \leq R^* + R/1000 \leq 2R/1000 \leq R/100,$$

as desired.  $\square$

### 5.2.8 Concavity

It is not hard to see that the score function is concave.

**Corollary 5.22** (COARSE-SDP concavity). *Let  $f(y)$  denote the value of COARSE-SDP for  $y$ . Then  $f$  is concave.*

*Proof.* Direct result of Lemma B.2.  $\square$

### 5.2.9 Accuracy Analysis of the Exponential Mechanism

Given that we have established bounded sensitivity (Lemma 5.17) and that the set of good solutions is sufficiently dense (Lemma 5.21), we can apply the volume-based exponential mechanism (Theorem 3.1).

**Lemma 5.23** (guarantee of the exponential mechanism for COARSE-SDP). *Let  $f(y)$  denote the value of COARSE-SDP for  $y$ . The exponential mechanism in Algorithm 2 with score function  $f$ , and privacy budget  $\varepsilon/m$  returns  $y$  such that*

$$f(y) \geq (1 - \alpha)n - \frac{2m}{\varepsilon} (d \log(2000) + \log(m/\beta)) ,$$

with probability  $1 - \beta/m$ . Specifically, if

$$n \geq \frac{2m}{\alpha\varepsilon} (d \log(2000) + \log(m/\beta)) ,$$

then

$$f(y) \geq (1 - 2\alpha)n ,$$

with probability  $1 - \beta/m$ . We say the exponential mechanism has been successful if the above inequality holds.

### 5.2.10 Runtime Analysis of the Exponential Mechanism

At this point, we have established bounded sensitivity (Lemma 5.17), Lipschitzness (Lemma 5.20), and concavity (Corollary 5.22). These properties suffice to apply the efficient sampling algorithm of Theorem 3.2.

**Lemma 5.24** (runtime of the exponential mechanism). *There exists a sampling scheme for the exponential mechanism in Algorithm 2 that takes time  $\text{poly}(n, d, \log R, 1/\varepsilon)$ .*

Note that, since the function is  $L = n\sqrt{d}/R$ -Lipschitz and since we are running the algorithm over the ball of radius  $R + R/1000$  (i.e.,  $\text{diam}(\mathcal{C}) \leq R + R/1000$ ), we have that  $L \text{diam}(\mathcal{C}) = O(n\sqrt{d})$  thus making the running time polynomial in  $n$  and  $d$ , but not in  $R$ .

## 5.3 Robustness

To conclude, we note that both of our coarse estimation algorithms are robust to contamination.

**Remark 5.25** (Robustness of Coarse Estimation). *For both coarse estimation algorithms (i.e., the coordinate-wise approach, Algorithm 1, and the SoS exponential mechanism approach, Algorithm 2), the SCORE function has sensitivity 1. Moreover, there exists a constant  $\eta_0$  (say  $\eta_0 \leq 0.01$ ), such that if SCORE is perturbed by  $\eta n$ ,  $\eta \leq \eta_0$ , all of our arguments still hold. Therefore, there exists a universal constant  $\eta_0$  for which our algorithm has the same guarantees if  $\eta$ -fraction of the samples are corrupted, where  $\eta \leq \eta_0$ .*

## 6 Fine Estimation

From Section 5, we know how to find an estimate of the mean up to distance  $\mathcal{O}(\sqrt{d})$ . In this section we prove the following theorem. Putting the following theorem and our theorem for coarse estimation (Theorem 5.1) together, gives us an efficient algorithm for privately estimating the mean of a  $d$ -variate distribution with bounded mean and covariance up to error  $\alpha$ .

**Theorem 6.1** (high dimensional fine mean estimation via SoS exponential mechanism). *Suppose  $\mathcal{D}$  is a  $d$ -dimensional distribution with mean  $\mu$  and covariance  $\Sigma$ , where  $\|\Sigma\|_2 \leq 1$ . Moreover, suppose that an initial estimate  $\tilde{\mu}_0$  is given such that  $\|\mu - \tilde{\mu}_0\|_2 \leq \mathcal{O}(\sqrt{d})$ . Then there exists a polynomial time  $\varepsilon$ -DP algorithm FINE-ESTIMATION that takes  $n$  samples from  $\mathcal{D}$  and outputs  $\tilde{\mu}^*$  such that  $\|\tilde{\mu}^* - \mu\|_2 \leq \alpha$ , with probability  $1 - \beta$ , and sample complexity*

$$n = \tilde{\mathcal{O}}\left(\frac{d + \log(1/\beta)}{\alpha^2 \varepsilon}\right).$$

More specifically,  $\tilde{\mathcal{O}}$  is hiding the following lower order logarithmic factors:  $\log d$ , and  $\log(1/\alpha)$

*Proof.* Applying Theorem 6.12 with  $m = 1/\alpha^2$ , and noting that Assumption 6.9 holds with probability  $1 - \beta$ , due to Lemma 6.8.  $\square$

Suppose  $\mathcal{D}$  is a  $d$ -dimensional distribution with mean  $\mu$ , and covariance  $\Sigma$  such that  $\|\Sigma\|_2 \leq 1$ . Our goal is to privately estimate the mean of this distribution up to accuracy  $\alpha$  with probability  $1 - \beta$ . Moreover, suppose that we have access to an initial estimate of the mean such that  $\|\mu - \tilde{\mu}_0\|_2 \leq \mathcal{O}(\sqrt{d})$ . We build our algorithm on top of the recent work in efficient high-dimensional mean estimation where the high-dimensional median of means technique is used [CFB19, Hop20, LM19b]. As opposed to non-biased estimators that have  $\mathcal{O}(\frac{d}{\beta\alpha^2})$  sample complexity, these estimators achieve the  $\mathcal{O}(\frac{d + \log(1/\beta)}{\alpha^2})$ , or the *sub-Gaussian rate* sample complexity.

In this technique, we take  $n = mk$  samples, and divide them into  $k$  bins  $\{\mathcal{B}\}_{i=1}^k$  each having  $m$  elements. Then we compute the means of each of these bins and name them  $Z_i$ 's. A key observation then is that if  $k \geq \max\{d, \Omega(\log(1/\beta))\}$ , then with probability  $1 - \beta$  over the randomness of the samples, the true mean,  $\mu$  is  $(0.99, \mathcal{O}(\sqrt{1/m}))$ -central.

**Definition 6.2** ( $(\rho, r)$ -centrality [LM19b, Hop20]). *Suppose  $Z_1, \dots, Z_k \in \mathbb{R}^d$ . A point  $x \in \mathbb{R}^d$  is  $(\rho, r)$ -central, if and only if for all unit vectors  $v$  we have*

$$|\{i \mid \langle Z_i - x, v \rangle \leq r\}| \geq \rho k$$

In other words, with high probability, for any direction  $v$ , if we center  $Z_i$ 's at the point  $\mu$ , and project them onto the direction  $v$ , a large fraction of the projections will fall in a radius of  $\mathcal{O}(\sqrt{1/m})$  of the mean. This can be viewed as  $\mu$  being a median in every direction.

Now note that if we can find some other point  $\hat{\mu}$  such that  $\|\hat{\mu} - \mu\|_2 \leq \mathcal{O}(\sqrt{1/m})$ , by setting  $m = 1/\alpha^2$  we may obtain the sub-Gaussian rate as desired. Suppose there exists another point  $\tilde{\mu}$  which is also  $(0.1, \mathcal{O}(\sqrt{1/m}))$ -central, then by choosing  $v$  along the direction of  $\mu - \tilde{\mu}$ , and applying the pigeonhole principle and the triangle inequality over the projections of  $Z_i$ 's in this direction, we may conclude that  $\|\tilde{\mu} - \mu\|_2 \leq \mathcal{O}(\sqrt{1/m})$ . Another useful observation is that if  $\|\tilde{\mu} - \mu\|_2 \leq \mathcal{O}(\sqrt{\frac{1}{m}})$ , then  $\tilde{\mu}$  is a central point. Therefore, we arrive at the following conclusion: assuming  $\mu$  is central; then a point  $\tilde{\mu}$  is central, if and only if it is close to  $\mu$ . This is the key observation that helps us characterize the points that are close to  $\mu$ . One can think of this centrality property as the “median” part of the “median of means” approach.

Therefore, answering the following question, gives us an algorithm for estimating the mean with sub-Gaussian rates.

**Main Problem** Suppose  $Z_1, \dots, Z_k \in \mathbb{R}^d$  are given, and there exists a point  $\mu \in \mathbb{R}^d$  which is a central point for  $Z_1, \dots, Z_k$ . How can we find a central point, both privately and efficiently?



Suppose we did not care about either privacy or efficiency. Similar to [CFB19], our goal is to start off from an estimate  $\tilde{\mu}$  and move towards a center  $\mu$ . If we were close to  $\mu$ , from the discussion above,  $\tilde{\mu}$  would have been a center. Therefore, one way to check whether or not we have reached our destination is by checking if  $\tilde{\mu}$  is central. What if  $\tilde{\mu}$  were far from the center  $\mu$  — which direction would move us towards  $\mu$ ? If  $\tilde{\mu}$  is far from the center, then  $\tilde{\mu}$  is not central and moreover, there exists some direction  $v$  such that

$$|\{i \mid \langle Z_i - \tilde{\mu}, v \rangle \geq 0.99 \|\mu - \tilde{\mu}\|_2\}| \geq 0.9k.$$

This direction  $v$  certifies that  $\tilde{\mu}$  is not central. Now a key observation is that if  $v$  certifies non-centrality, then  $v$  points towards  $\mu$ , i.e.

$$\left\langle v, \frac{\mu - \tilde{\mu}}{\|\mu - \tilde{\mu}\|_2} \right\rangle \geq \Omega(1).$$

Now, if we move in the direction of  $v$  (which certifies non-centrality) with step-size proportional to the distance we are from  $\mu$  (i.e.,  $\|\tilde{\mu} - \mu\|_2$ ), we can move towards  $\mu$  at a linear rate. Doing this for a number of steps which is logarithmic in the initial distance gives us a point which is central with respect to  $Z_i$ 's and has distance  $\mathcal{O}(\sqrt{1/m})$  to  $\mu$ .

In order to employ this approach both efficiently and privately we need to execute the following tasks efficiently and privately:

1. Check whether or not the current estimate  $\tilde{\mu}$  is central.
2. Find an estimate of the distance to the central point or  $\|\tilde{\mu} - \mu\|_2$ .
3. Find a direction  $v$  that certifies the non-centrality of  $\tilde{\mu}$ .

Even setting the concern of privacy aside for the moment, it is not clear how to efficiently check whether or not the current estimate  $\tilde{\mu}$  is central. One approach involves constructing a net over the unit sphere  $\mathcal{S}^{d-1}$  and counting the number of  $Z_i$ 's that fall far for each direction in the unit sphere. However, doing so would take exponential time as the number of points on the net grows exponentially in  $d$ . In order to perform this efficiently, [CFB19] uses the standard SDP relaxation of the following quadratic program, which finds the direction that has most of  $Z_i$ 's at distance at least  $r$  from the current estimate  $\tilde{\mu}$ . Note that a point  $x$  is  $(\rho, r)$ -central if and only if the following quadratic program has a low value.

**Definition 6.3** (quadratic optimization problem [Hop20, CFB19]). *Let  $Z_1, \dots, Z_k, \tilde{\mu} \in \mathbb{R}^d$ ,  $r > 0$ . Let  $\text{QUAD}(\tilde{\mu}, r, Z)$  be the following quadratic program.*

$$\begin{aligned} \text{QUAD}(\tilde{\mu}, r, Z) &:= \max_{v, b} \sum_{i=1}^k b_i^2 \\ &s.t. \quad b_i^2 = b_i \quad \forall i \\ &\quad \sum_{i=1}^d v_i^2 = 1 \\ &\quad b_i^2 r \leq \langle Z_i - \tilde{\mu}, b_i v \rangle \quad \forall i \end{aligned}$$

The following is the standard SDP relaxation of the above quadratic program.

**Definition 6.4** (SDP Relaxation [Hop20, CFB19]). Let  $Z_1, \dots, Z_k, \tilde{\mu} \in \mathbb{R}^d$ ,  $r > 0$ . Let  $\text{SDP}(\tilde{\mu}, r, Z)$  be the following semi-definite program.

$$\begin{aligned} \text{SDP}(\tilde{\mu}, r, Z) := & \max_{v, b, B, U, W} \text{Tr}(B) \\ \text{s.t.} & \begin{bmatrix} 1 & b^\top & v^\top \\ b & B & W \\ v & W^\top & V \end{bmatrix} \succcurlyeq 0 \\ & B_{ii} = b_i \quad \forall i \\ & \text{Tr}(V) = 1 \quad \forall i \\ & B_{ii} \cdot r \leq \langle Z_i - \tilde{\mu}, W_i \rangle \quad \forall i \end{aligned}$$

Now based on the above SDP relaxation, we may define a new notion of centrality.

**Definition 6.5** ( $(\rho, r)$ -SDP-centrality [Hop20]). A point is  $(\rho, r)$ -SDP-central if and only if the SDP relaxation's value for that point is less than  $1 - \rho$ .

The main advantage SDP-centrality has over the previous notion of centrality is that it can be computed efficiently. It can be proven that SDP-centrality has the same good properties that the previous definition of centrality provides:

1. The true mean  $\mu$  is  $(0.999, \mathcal{O}(\sqrt{1/m}))$ -SDP-central with high probability (Lemma 6.8).
2. Under the assumption that  $\mu$  is  $(0.999, \mathcal{O}(\sqrt{1/m}))$ -SDP-central, we have that  $\tilde{\mu}$  is  $(0.1, \mathcal{O}(\sqrt{1/m}))$ -SDP-central if and only if  $\|\mu - \tilde{\mu}\|_2 \leq \mathcal{O}(\sqrt{1/m})$  (a corollary of the proof of Theorem 6.13, Lemma 4 of [CFB19])
3. The solution to the SDP relaxation can certify the non-SDP-centrality of points which are far from a SDP-center. Moreover, there exists a rounding scheme that obtains a direction  $v$  from this solution such that  $\langle v, \frac{\mu - \tilde{\mu}}{\|\mu - \tilde{\mu}\|} \rangle \geq \Omega(1)$ .<sup>2</sup>

These properties of the SDP relaxation, together with its computational efficiency, yield a computationally efficient algorithm for estimating the mean up to  $\mathcal{O}(\sqrt{1/m})$  error.

Now that we have reviewed [CFB19]'s method, we describe how our *private* algorithm works. One naive approach to ensuring privacy would be to add Laplace noise to the direction after the rounding step. However, it is not clear how to prove sensitivity bounds in this case to guarantee pure differential privacy. In order to privately find the direction, we run the exponential mechanism over the unit ball.<sup>3</sup> If the score function of the exponential mechanism for a given vector  $y$  is

$$\text{SCORE}(y) = \sum_{i=1}^k \mathbb{1}[\langle Z_i - \tilde{\mu}, y \rangle \geq r]$$

and we assume that the exponential mechanism always returns the  $y$  with the highest score, the exponential mechanism would return a direction  $y$  such that  $\langle y, \mu - \tilde{\mu} \rangle \geq \Omega(1)$ . Moreover, the sensitivity of the score function is only 1. However, naïve implementations of the exponential mechanism would not be computationally efficient, running in time exponential in  $d$ . To alleviate this issue,

<sup>2</sup>[CFB19] shows that taking the top eigenvector of  $V$  in Definition 6.4 is one such rounding scheme. In Theorem 6.23, we show that the simple rounding scheme of taking  $v$  itself provides the same guarantees.

<sup>3</sup>To be precise, in our final algorithm this is not exactly the unit ball. We run the exponential mechanism over a ball of radius  $1 - \Theta(1)$ . This helps us ensure Definition 4.3 or Lipschitzness.

we recall the efficient sampling algorithm of [BST14]. Their results say that if a score function is concave and Lipschitz, and is defined over a bounded and convex domain, then appropriate techniques from the literature on sampling log-concave distributions allow us to efficiently implement the exponential mechanism. Meanwhile, this score function may not even be continuous.

To define a score function  $\text{SCORE}(y)$  that has the mentioned properties, we use the SDP relaxation mentioned in Definition 6.4, and intersect it with a new condition  $v = y$  to define SDP-VAL (see Definition 6.7). In order to show that sampling from this SCORE function can be done efficiently, we show that it is concave (Corollary 6.29) and Lipschitz (Lemma 6.28). Moreover, in order to show that with high probability the exponential mechanism returns a high scoring vector, we show that the score function has sensitivity 1 and that the volume of the high scoring points is large (Corollary 6.26, Section 6.5.8). It remains to show that given a feasible solution  $X(1, b, v, B, V)$  to the SDP program defined in Definition 6.4 with high value,  $\langle v, \|\mu - \tilde{\mu}\|_2 \rangle \geq \Omega(1)$ . This guarantees that if the exponential mechanism returns a high-scoring vector, then that vector points towards  $\mu$ . Note that previous works use a slightly different rounding scheme. In [CFB19], they show that the top eigenvector of the sub-matrix  $V$  of  $X(1, b, v, B, V)$  has the said property; however we show that a simpler rounding scheme (alluded to above) also gives our desired guarantees (Theorem 6.23). For further details on gradient estimation, see Section 6.5.

It remains to address the other two components of the algorithm: determining when to stop, and estimating the distance. For the former, the problem would be trivial without privacy: one could simply check the value of the SDP with  $r = \Theta(\sqrt{1/m})$ . If the value is low, we would be close, otherwise, we would have been far. Our private approach is similar, but we also add noise to the value of the SDP with Laplace mechanism to ensure privacy. We prove the SDP value has sensitivity 1, in order to bound the magnitude of the Laplace noise. Further discussion on how to determine when to stop appears in Section 6.3.

Finally, for distance estimation sans privacy, a binary search suffices to select an  $r$ : find the maximum  $r$  such that for this choice of  $r$ , the SDP has value larger than  $0.9k$ . It can be proven that choosing  $r$  this way gives an approximation of the distance to the center [CFB19]. However, it is not clear how to directly make this approach private. Therefore, we instead find some  $r$  for which the SDP has value between  $0.9k$  and  $0.95k$  via a private binary search. Note that if we are far enough from the center, and if we are given a good coarse estimate, both of these values should be feasible, which is ensured by the guarantee of the stopping step and having access to a coarse estimate. Since the value of SDP is a decreasing function in  $r$  and we are sufficiently far from the center, it can be proven that if the SDP has value between  $0.9k$  and  $0.95k$  for  $r$ , then  $r$  is an approximation of the distance to the center. Full discussion on the private binary search procedure and private distance estimation appears in Section 6.4.

**Structure.** First we define a new quadratic program and a new SDP that we use to define a score function for the exponential mechanism that helps us find the gradient in our algorithm. Then we review some useful lemmas and assumptions from [CFB19] in Section 6.1.2. After that we propose the main algorithm and the state the guarantees it provides in Section 6.2. As discussed above, our algorithm has three parts: estimating whether it should stop or not, estimating the distance, and estimating the direction to the true mean. We present our algorithms, guarantees and proofs, for these three tasks in Section 6.3, Section 6.4, and Section 6.5 respectively. In Section 6.6 we show that after sufficient number of rounds of the gradient descent our algorithm obtains a point close to the true mean. Finally, in Section 6.7, we explain why our algorithm is robust under constant fraction corruption in the samples.

## 6.1 Preliminaries

### 6.1.1 Definitions

In order to run the exponential mechanism we need to define a score function over the set of directions. In order to do so, we define the following programs based on the previous definitions (Definition 6.3, Definition 6.4). These programs are the intersection of the above programs with the new condition,  $v = y$ . These new programs can be interpreted as: for a given direction  $v$ , how many of the  $Z_i$ 's are at distance at least  $r$  from the current estimate  $\tilde{\mu}$ , in the direction  $v$ ?

**Definition 6.6** (QUAD-VAL). *Let  $Z_1, \dots, Z_k, \tilde{\mu}, y \in \mathbb{R}^d$ ,  $r > 0$ . Let  $\text{QUAD-VAL}(y; \tilde{\mu}, r, Z)$  be the following quadratic program.*

$$\begin{aligned} \text{QUAD-VAL}(y; \tilde{\mu}, r, Z) &:= \max_{v, b} \sum_{i=1}^k b_i^2 \\ \text{s.t.} \quad &b_i^2 = b_i \quad \forall i \\ &v_i = y_i \quad \forall i \\ &b_i^2 r \leq \langle Z_i - \tilde{\mu}, b_i v \rangle \quad \forall i \end{aligned}$$

Unlike the previous SDP (Definition 6.4), the following SDP is *not* a standard SDP relaxation of the above quadratic program. This is because, in order to be able to run the exponential mechanism efficiently, we need to define our score function over the unit ball as opposed to the unit sphere; hence,  $\|y\|_2$  may not be equal to 1. However, we show that  $\text{SDP-VAL}(y)$  is larger than  $\text{QUAD-VAL}(y)$ , in Lemma A.5.

**Definition 6.7** (SDP-VAL). *Let  $Z_1, \dots, Z_k, \tilde{\mu}, y \in \mathbb{R}^d$ ,  $r > 0$ . Let  $\text{SDP-VAL}(y; \tilde{\mu}, r, Z)$  be the following semi-definite program.*

$$\begin{aligned} \text{SDP-VAL}(y; \tilde{\mu}, r, Z) &:= \max_{v, b, B, W, V} \text{Tr}(B) \\ \text{s.t.} \quad &\begin{bmatrix} 1 & b^\top & v^\top \\ b & B & W \\ v & W^\top & V \end{bmatrix} \succcurlyeq 0 \\ &v_i = y_i \quad \forall i \\ &B_{ii} = b_i \quad \forall i \\ &\text{Tr}(V) = 1 \\ &B_{ii} \cdot r \leq \langle Z_i - \tilde{\mu}, W_i \rangle \quad \forall i \end{aligned}$$

*Note that the above semi-definite program can also be viewed as the following convex program, where the maximization is over degree-2 pseudo-distributions in indeterminates  $v_1, \dots, v_d, b_1, \dots, b_k$ . In our proofs, we use the two definitions interchangeably.*

$$\begin{aligned} \text{SDP-VAL}(y; \tilde{\mu}, r, Z) &:= \max_{\tilde{\mathbf{E}}} \sum_{i=1}^k b_i \\ \text{s.t.} \quad &\tilde{\mathbf{E}} v_i = y_i \quad \forall i \\ &\tilde{\mathbf{E}} \text{ satisfies } b_i = b_i^2 \quad \forall i \\ &\tilde{\mathbf{E}} \text{ satisfies } \|v\|^2 = 1 \\ &\tilde{\mathbf{E}} \text{ satisfies } b_i^2 r \leq \langle Z_i - \tilde{\mu}, b_i v \rangle \quad \forall i \end{aligned}$$

### 6.1.2 Useful Lemmas

In this subsection we state some useful lemmas regarding the solutions of the quadratic program and the SDP defined in Definition 6.3, and Definition 6.4. These lemmas are due to [CFB19]. We state a more general version of them due technical reasons, however the proofs remain similar.

The following lemma states that if we are at the true mean, then with high probability, the semidefinite program defined in Definition 6.4, cannot have a high score. In the quadratic program's language, this corresponds to a large fraction of  $Z_i$ 's being close to the true mean when projected onto any given direction.

**Lemma 6.8** (Lemma 7 in [CFB19]). *Let  $M \geq 20$ , and  $\mathcal{Y} = (Y_1, \dots, Y_k) \in \mathbb{R}^{k \times d}$  be  $k$  i.i.d. random vectors with mean  $\mu$  and covariance  $\Lambda$  and let  $\mathcal{S}$  denote the set of feasible solutions of the SDP( $\mu, r, \mathcal{Y}$ ) in Definition 6.4. Then, we have for  $r \geq 5M(\sqrt{\text{Tr } \Lambda/k} + \sqrt{\|\Lambda\|})$  and  $k \geq 8M^2 \log 1/\beta$*

$$\max_{(v,b,B,U,W) \in \mathcal{S}} \text{Tr } B \leq \frac{k}{M},$$

with probability at least  $1 - \beta$ . More specifically, for  $r \geq 10M\sqrt{\|\Lambda\|_2}$  if  $k \geq 8M^2 \log 1/\beta$ , and  $k \geq d$ ,

$$\max_{(v,b,B,U,W) \in \mathcal{S}} \text{Tr } B \leq \frac{k}{M},$$

with probability at least  $1 - \beta$ .

Therefore, we may assume that with high probability, the following assumption holds. In [CFB19], they state the above lemma and the following assumption with  $M = 20$ . In our proofs we require a larger value for  $M$ . For the sake of tidiness, we do not set a value for  $M$  in the following assumptions and we state them for general  $M$ . However, in our proofs we only require the following assumptions to hold for  $M = 1000$ .

**Assumption 6.9** (Assumption 2 in [CFB19]). *Suppose  $r^* = 2\sqrt{k/n} = 2\sqrt{1/m}$ . For the bucket means  $Z = (Z_1, \dots, Z_k)$  let  $\mathcal{S}_r$  denote the set of feasible solutions for SDP( $\mu, r, Z$ ). Then, for all  $r \geq 5Mr^*$ ,*

$$\max_{v,b,B,W,V \in \mathcal{S}_r} \sum_{i=1}^k \text{Tr } B \leq \frac{k}{M}$$

The above assumption is a strengthening of the following assumption.

**Assumption 6.10** (Assumption 1 in [CFB19]). *For the bucket means,  $Z = (Z_1, \dots, Z_k)$ , we have:*

$$\forall v \in \mathbb{R}^d, \|v\|_2 = 1 \implies |\{i : \langle Z_i - \mu, v \rangle \geq 5Mr^*\}| \leq \frac{k}{M}$$

The following lemma is about the feasible solutions to the SDP problem that have high values.

**Lemma 6.11** (Lemma 3 in [CFB19]). *Let us assume Assumption 6.9. Let*

$$X = \begin{bmatrix} 1 & b^\top & v^\top \\ b & B & W \\ v & W^\top & V \end{bmatrix} \in \mathbb{R}^{(k+d+1) \times (k+d+1)}$$

be a PSD matrix. Moreover, suppose that

$$B_{i,i} = b_i, \quad \text{Tr}(V) = 1, \quad \text{Tr}(B) \geq \psi k.$$

Then, there is a set of at least  $(\psi - 1/M)k$  indices  $\mathcal{T}$  such that for all  $i \in \mathcal{T}$ :

$$\langle Z_i - \mu, W_i \rangle < 5Mr^*B_{i,i},$$

and a set of at least  $(\psi - \psi')/(1 - \psi')k$  indices  $\mathcal{R}$  such that for all  $j \in \mathcal{R}$ , we have  $B_{j,j} \geq \psi'$ , where  $0 \leq \psi' \leq \psi$ .

## 6.2 Main Algorithm and Theorem

In this section we state our main algorithm and theorem. As discussed before, our main algorithm is a gradient descent style algorithm and consists of three parts: HALT-ESTIMATION, DISTANCE-ESTIMATION, GRADIENT-ESTIMATION. We discuss these algorithms and their guarantees in Section 6.3, Section 6.4, and Section 6.5 respectively.

---

### Algorithm 3 FINE-ESTIMATION

---

**Input:** Parameters: Data points  $X_1, \dots, X_n \in \mathbb{R}^d$ , number of bins  $k$ , number of iterations  $T$ , initial estimate  $\tilde{\mu}_0$ .

**Operation:** Step-size  $\eta = 0.075$

Split the data in  $k$  bins  $\{\mathcal{B}_i\}_{i=1}^k$ , each having  $m$  members. For all  $1 \leq i \leq k$ , let  $Z_i = \text{MEAN}(\mathcal{B}_i)$ .

For  $1 \leq t \leq T$  do:

1. Let  $h_t = \text{HALT-ESTIMATION}(Z, \tilde{\mu}_{t-1})$
2. If  $h_t$ , then return  $\tilde{\mu}_{t-1}$ .
3. Let  $d_t = \text{DISTANCE-ESTIMATION}(Z, \tilde{\mu}_{t-1})$
4. Let  $g_t = \text{GRADIENT-ESTIMATION}(Z, \tilde{\mu}_{t-1}, d_t)$
5. Let  $\tilde{\mu}_t = \tilde{\mu}_{t-1} + \eta d_t g_t$

**Output:**  $\tilde{\mu}_T$

---

The following theorem is the guarantee that our algorithm provides, under the assumption that Assumption 6.9 holds, and an initial estimate is given. This theorem together with Lemma 6.8, gives us Theorem 6.1.

**Theorem 6.12** (FINE-ESTIMATION). *Let  $Z_1, \dots, Z_k, \tilde{\mu}_0, \mu \in \mathbb{R}^d$   $r^* = 2\sqrt{k/n} = 2\sqrt{1/m}$ . Suppose  $\|\tilde{\mu}_0 - \mu\|_2 \leq C_0\sqrt{d}$  where  $C_0$  is a universal constant,  $M = 1000$ , and Assumption 6.9 holds. Then there exists an efficient  $\varepsilon$ -DP algorithm FINE-ESTIMATION that returns  $\tilde{\mu}^*$  such that if  $k \geq \tilde{\mathcal{O}}((d + \log(1/\beta))/\varepsilon)$ , then  $\|\tilde{\mu}^* - \mu\|_2 \leq \mathcal{O}(r^*)$ , with probability  $1 - \beta$ . Note that  $\tilde{\mathcal{O}}$  is hiding the following lower order logarithmic factors:  $\log d$ , and  $\log m$ .*

*Proof.* Putting together Theorem 6.13 (HALT-ESTIMATION), Theorem 6.14 (DISTANCE-ESTIMATION), Theorem 6.22 (GRADIENT-ESTIMATION), and Theorem 6.39 (gradient descent), and setting  $T = \log(dm) \geq \log(d/r^*)$ .  $\square$

### 6.3 Halt Estimation

If we did not care about privacy, in order to know whether we are close to  $\mu$  or not, and we had a current estimate  $\tilde{\mu}$ , we could just check if SDP has score smaller than  $0.9k$  for a small enough distance choice or not. If that was the case,  $\tilde{\mu}$  would have been close enough to  $\mu$ . In this section we privatize this condition by adding Laplace noise to the value of SDP, and argue that this algorithm provides privacy guarantees and high accuracy with high probability.

---

#### Algorithm 4 HALT-ESTIMATION

---

**Input:** Parameters: Data points  $Z_1, \dots, Z_k \in \mathbb{R}^d$ , current estimate  $\tilde{\mu}$ .

**Operation:** Let  $M = 1000, N = 4M(5M + 1) + 10M$

Let  $H = \mathbb{1}[\text{SDP}(\tilde{\mu}, Nr^*, Z) + \mathcal{L}(T/\varepsilon) \leq 0.91k]$ , where  $\mathcal{L}$  is the Laplace distribution.

**Output:**  $H$

---

**Theorem 6.13** (HALT-ESTIMATION). *Let  $Z_1, \dots, Z_k, \tilde{\mu} \in \mathbb{R}^d$ ,  $r^* = 2\sqrt{k/n}$ ,  $M \geq 1000$ ,  $N = 4M(5M + 1) + 10M$ . Suppose  $k \geq 100T \log(T/\beta)/\varepsilon$ , and that Assumption 6.9 holds. There exists an  $\varepsilon/T$ -DP algorithm HALT-ESTIMATION that succeeds with probability  $1 - \beta/T$ . Moreover, in the regime where this algorithms succeeds we have*

$$H = 0 \implies \|\mu - \tilde{\mu}\|_2 \geq M(5M + 1)r^*,$$

and

$$H = 1 \implies \|\mu - \tilde{\mu}\|_2 \leq Nr^* + 5Mr^*.$$

*Proof.* Since SDP has sensitivity 1 (Corollary 6.26), from the guarantees of the Laplace mechanism we know that Algorithm 4 is  $\varepsilon/T$ -DP and with probability  $1 - \beta/T$ , it succeeds i.e.  $\text{SDP}(\tilde{\mu}, Nr^*, Z) + \mathcal{L}(T/\varepsilon) \in [\text{SDP}(\tilde{\mu}, Nr^*, Z) - T \log(T/\beta)/\varepsilon, \text{SDP}(\tilde{\mu}, Nr^*, Z) + T \log(T/\beta)/\varepsilon]$ . Since  $k \geq 100T \log(T/\beta)$ , with probability  $1 - \beta/T$

$$\text{SDP}(\tilde{\mu}, Nr^*, Z) + \mathcal{L}(T/\varepsilon) \in [\text{SDP}(\tilde{\mu}, Nr^*, Z) - 0.1k, \text{SDP}(\tilde{\mu}, Nr^*, Z) + 0.1k].$$

Therefore if we are in the success regime,  $H = 0$ , means that

$$\text{SDP}(\tilde{\mu}, Nr^*, Z) \geq 0.90k,$$

and  $H = 1$ , means that

$$\text{SDP}(\tilde{\mu}, Nr^*, Z) \leq 0.92k.$$

Now by Corollary 6.24, if we are in the success regime

$$H = 0 \implies \|\mu - \tilde{\mu}\|_2 \geq \frac{0.7Nr^* - 5Mr^*}{2} \geq M(5M + 1)r^*,$$

where the last inequality comes from  $N = 4M(5M + 1) + 10M$ . Now suppose that we are in the success regime and  $H = 1$ . Then  $\text{SDP}(\tilde{\mu}, Nr^*, Z) \leq 0.92k$ , which implies that  $\text{QUAD}(\tilde{\mu}, Nr^*, Z) \leq 0.92k$ , since SDP is a relaxation of QUAD. Therefore for any  $v \in \mathbb{R}^d$ , where  $\|v\|_2 = 1$  we have that

$$|\{i \mid \langle Z_i - \tilde{\mu}, v \rangle \leq Nr^*\}| \geq 0.08k.$$

On the other hand, from Assumption 6.10 (which is a result of Assumption 6.9), we know that for any  $v' \in \mathbb{R}^d$ , where  $\|v'\|_2 = 1$  we have that

$$|\{i \mid \langle Z_i - \mu, v' \rangle \leq 5Mr^*\}| \geq (1 - \frac{1}{M})k.$$

Let  $v = (\mu - \tilde{\mu})/\|\mu - \tilde{\mu}\|_2$ ,  $v' = -v$ . Then  $M \geq 1000$  and  $0.08 + (1 - 1/M) > 1$ , there exists some index  $j$  which is in the intersection of the above sets. Therefore for this  $j$  we have that

$$\langle Z_j - \mu + \mu - \tilde{\mu}, v \rangle \leq Nr^*,$$

therefore

$$\|\mu - \tilde{\mu}\|_2 \leq Nr^* + \langle Z_j - \mu, -v \rangle \leq Nr^* + 5Mr^*,$$

as desired.  $\square$

## 6.4 Distance Estimation

Suppose we have a current estimate  $\tilde{\mu}$ , and  $\mu$  is a central point. Furthermore suppose that  $\tilde{\mu}$  and  $\mu$  are sufficiently far from each other. In this section our goal is to find an estimate of the distance  $\|\tilde{\mu} - \mu\|_2$ . If we did not care about privacy, we could look for an approximation of the distance to a central point by finding the maximum  $r$  such that SDP obtains value at least  $0.9k$  via a binary search up to arbitrary accuracy. However, it's not clear how to privatize this approach. Instead, we look for an  $r$  for which SDP obtains a value between  $0.91k$  and  $0.94k$ , via a private binary search (Theorem 6.15, Lemma 6.16). If  $r$  has this property, then we can use the monotonicity of SDP in  $r$  to argue that  $r$  is an approximation of the distance the central point (Lemma 6.17, Lemma 6.18).

### 6.4.1 Algorithm

---

#### Algorithm 5 DISTANCE-ESTIMATION

---

**Input:** Parameters: Data points  $Z_1, \dots, Z_k \in \mathbb{R}^d$ , current estimate  $\tilde{\mu}$ .

**Operation:** Let  $r^* = 2\sqrt{k/n}$ ,  $S = \log(1.14C_0\sqrt{d}/r^*)$ ,  $s = 0.92k$ ,  $e = 0.93k$ .

Do the following binary search for  $S$  rounds over  $[0, 1.14C_0\sqrt{d}]$ , with  $r_0$  being the middle point

1. Let  $u = \text{SDP}(\tilde{\mu}, r_t, Z) + N_i$ , where  $N_i \sim \mathcal{L}(TS/\varepsilon)$ .
2. Let  $u$  is larger than  $ek$  move left, it is less than  $ek$  move right, if it is neither, move in an arbitrary direction.

**Output:**  $r_T$ .

---

### 6.4.2 Main Theorem

**Theorem 6.14** (DISTANCE-ESTIMATION). *Let  $Z_1, \dots, Z_k, \tilde{\mu}, \mu \in \mathbb{R}^d$ . Suppose  $r^* = 2\sqrt{k/n}$ ,  $m = n/k$ ,  $\|\mu - \tilde{\mu}\|_2 \geq M(5M + 1)r^*$ ,  $k \geq 100ST \log(ST/\beta)/\varepsilon$ ,  $M \geq 100$ , and that Assumption 6.9 holds. Moreover, suppose that  $\|\mu - \tilde{\mu}\|_2 \leq C_0\sqrt{d}$ , where  $C_0$  is a universal constant. Then there exists*



a universal constant  $C_2$  and a  $\varepsilon/T$ -dp algorithm `DISTANCE-ESTIMATION` using  $S$  rounds of binary search that returns  $\hat{d}$  such that

$$0.99\|\mu - \tilde{\mu}\|_2 \leq \hat{d} \leq 1.15\|\mu - \tilde{\mu}\|_2,$$

with probability  $1 - \beta/T$ , where  $S = C_2(\log m + \log d)$ . We say that `DISTANCE-ESTIMATION` has been successful if the above guarantee holds.

*Proof.* See Section 6.4.4. □

### 6.4.3 Private Binary Search

The following theorem allows us to perform private binary search over a decreasing function, with low sensitivity.

**Theorem 6.15** (Private Binary Search). *Suppose  $f: [0, D] \times \mathcal{X} \rightarrow [0, k]$  is a decreasing function and has sensitivity 1 with respect to its second component. Moreover two values  $0 \leq s < e \leq 1$  are given. Let  $r_s = \operatorname{argmax}_{f(r) \geq sk - \Delta} r$ ,  $r_e = \operatorname{argmin}_{f(r) \leq ek + \Delta} r$ , then there exists an  $\varepsilon$ -DP algorithm using  $S$  rounds of binary search that returns  $x$  such that  $x \in [r_e - a, r_s + a]$  with probability  $1 - \beta$ , where  $\Delta = S \log(S/\beta)/\varepsilon$ , and  $S = \log(D/a)$ .*

*Proof.* Suppose  $x_i$  is the binary search parameter at step  $i$ , do the following binary search up to  $S$  steps over the domain  $[0, D]$  and in the end return  $x_S$ .

1. Let  $f_i = f(x_i)$ , and  $f'_i = f_i + N_i$ , where  $N_i \sim \mathcal{L}(S/\varepsilon)$ .
2. If  $f'_i$  is larger than  $ek$  move right, if it is less than  $sk$  move left, if it is neither, move in an arbitrary direction.

Now we need to prove privacy and accuracy guarantees for this algorithm.

**Privacy** The algorithm is  $\varepsilon$ -DP by basic composition. Note that each step is  $\varepsilon/S$ -DP.

**Accuracy** Note that for every  $t \geq 0$

$$\mathbb{P}[|N_i| \geq tS/\varepsilon] = \exp(-t).$$

and by union bound

$$\mathbb{P}[\max |N_i| \geq tS/\varepsilon] \leq S \exp(-t).$$

Let  $t = -\log(\beta/T)$ , since  $\Delta = S \log(S/\beta)/\varepsilon$ ,

$$\mathbb{P}[\max |N_i| \geq \Delta] \leq \beta.$$

Now with probability  $1 - \beta$ ,  $\forall i, N_i \leq \Delta$ . Suppose we are in this setting where all of the  $N_i$ 's are smaller than  $\Delta$ . We have three cases.

1. If  $x_i \in (r_s, D]$ , then  $f_i \in [0, sk - \Delta)$ , and  $f'_i \in [-\Delta, sk)$ , therefore, the binary search algorithm will choose left.
2. If  $x_i \in [r_e, r_s]$ , then  $f_i \in [sk - \Delta, ek + \Delta]$ , and  $f'_i \in [sk - 2\Delta, ek + \Delta]$ , therefore the binary search algorithm might choose right or left (arbitrary direction).
3. If  $x_i \in [0, r_e)$ , then  $f_i \in (ek + \Delta, k]$ , and  $f'_i \in (ek, k + \Delta]$ , the binary search algorithm will choose right.

Therefore this algorithm is performing Binary Interval Search for the interval  $[r_e, r_s]$  with accuracy parameter  $a$ . Therefore, by Lemma B.3, with probability  $1 - \beta$ , the algorithm returns a point  $x$ , such that  $x \in [r_e - a, r_s + a]$ . □

#### 6.4.4 Proof of Theorem 6.14

In this section we prove Theorem 6.14. This theorem is proven by putting together Lemma 6.16, Lemma 6.17, and Lemma 6.18.

**Lemma 6.16** (SDP Binary Search). *Let  $Z_1, \dots, Z_k, \tilde{\mu}, \mu \in \mathbb{R}^d$ . Suppose  $r^* = 2\sqrt{k/n}$ ,  $m = n/k$ ,  $\|\mu - \tilde{\mu}\|_2 \leq C_0\sqrt{d}$ ,  $\|\mu - \tilde{\mu}\|_2 \geq M(5M+1)r^*$ ,  $k \geq 100ST \log(ST/\beta)/\varepsilon$ ,  $M \geq 100$ ,  $C_1 = 1.14C_0$ , and that Assumption 6.9 holds. Let  $f(r) := \text{SDP}(\tilde{\mu}, r, Z)$ , where  $f : [0, C_1\sqrt{d}] \rightarrow \mathbb{R}$ . Let  $\rho_s = \arg\max_{f(r) \geq 0.91k} r$ ,  $\rho_e = \arg\min_{f(r) \leq 0.94k} r$ . Then there exists a universal constant  $C_2$  and a  $\varepsilon/T$ -dp algorithm using  $S$  rounds of binary search that returns  $\hat{r}$  such that  $\hat{r} \in [\rho_e - r^*, \rho_s + r^*]$  with probability  $1 - \beta/T$ , where  $S = C_2(\log m + \log d)$ . Moreover, if  $\|\mu - \tilde{\mu}\|_2 \leq M(5M+1)r^*$*

*Proof.* We want to apply the algorithm from Theorem 6.15. In order to do so, we must check that SDP is decreasing in  $r$  (Lemma A.1) and has sensitivity 1 (Corollary 6.26). Furthermore, note that  $\{r : f(r) \geq 0.91k, 0 \leq r \leq C_1\sqrt{d}\}$ , and  $\{r : f(r) \leq 0.94k, 0 \leq r \leq C_1\sqrt{d}\}$  are non-empty sets, which is true because of Corollary 6.19, and Corollary 6.32. Now we may apply Theorem 6.15 with

$$s \leftarrow 0.92k, \quad e \leftarrow 0.93k, \quad D \leftarrow C_1\sqrt{d}, \quad a \leftarrow r^* = 2\sqrt{k/n}.$$

Therefore,  $S = \log(D/a) = \log \sqrt{\frac{C_1 dn}{4k}}$ . Therefore, there exists a universal constant  $C_2$  such that

$$S \leq C_2(\log d + \log m).$$

It remains to bound  $\Delta$  by  $0.01k$ , which is true by the assumption.

$$\begin{aligned} \Delta &= ST \log(ST/\beta)/\varepsilon \\ &\leq 0.01k \end{aligned}$$

Therefore, using the algorithm from Theorem 6.15 we constructed an algorithm that returns  $\hat{r} \in [\rho_e + r^*, \rho_s - r^*]$  with probability  $1 - \beta/T$ , using  $\mathcal{O}(\log m + \log d)$  steps of the binary search, as desired.  $\square$

**Lemma 6.17** (lower bound for distance estimate). *Under the assumptions of Lemma 6.16,  $\hat{r} \geq \rho_e - r^*$  implies  $\hat{r} \geq (1 - 1/M)\|\tilde{\mu} - \mu\|_2$ .*

*Proof.* Let  $\Delta$  be the unit direction  $\tilde{\mu} - \mu$ , then by Assumption 6.10 (which is implied by Assumption 6.9),

$$|\{i : \langle Z_i - \mu, \Delta \rangle \geq 5Mr^*\}| \leq k/M.$$

Therefore if we take the direction  $-\Delta$ , we have for  $(1 - 1/M)k$  of the points that

$$\langle Z_i - \tilde{\mu}, -\Delta \rangle = \langle \tilde{\mu} - \mu + \mu - Z_i, \Delta \rangle \geq \|\tilde{\mu} - \mu\|_2 - 5Mr^*$$

Therefore  $\text{SDP}(\tilde{\mu}, \|\tilde{\mu} - \mu\|_2 - 5Mr^*, Z) \geq (1 - 1/M)k$ , since SDP is the standard relaxation of QUAD. Now note that by Lemma 6.16,  $\text{SDP-VAL}(\tilde{\mu}, \hat{r}, Z) \leq 0.94k$ ; therefore,

$$\text{SDP}(\tilde{\mu}, \|\tilde{\mu} - \mu\|_2 - 5Mr^*, Z) \geq (1 - 1/M)k \geq 0.95k > 0.94k \geq \text{SDP}(\tilde{\mu}, \rho_e, Z).$$

Now by monotonicity of SDP in the second parameter, we conclude that  $\rho_e \geq \|\tilde{\mu} - \mu\|_2 - 5Mr^*$ . Therefore

$$\hat{r} \geq \rho_e - r^* \geq \|\tilde{\mu} - \mu\|_2 - (5M+1)r^* \geq \left(1 - \frac{1}{M}\right)\|\tilde{\mu} - \mu\|_2,$$

as desired, where the final inequality comes from  $\|\tilde{\mu} - \mu\|_2 \geq M(5M+1)r^*$ .  $\square$

**Lemma 6.18** (upper bound for distance estimation). *Under the same assumptions as in Lemma 6.16,  $\hat{r} \leq \rho_s + r^*$  implies  $\hat{r} \leq 1.15\|\tilde{\mu} - \mu\|_2$ .*

*Proof.* The proof is similar to the proof in Lemma 5 from [CFB19]. We show that  $\text{SDP}(\tilde{\mu}, 1.14\|\tilde{\mu} - \mu\|_2, Z) \leq 0.9k$ . Suppose the contrary, then there exists some solution  $(b, v, B, W, V)$  that achieves value larger than  $0.9k$ . Let  $\mathcal{R}, \mathcal{T}$  be the sets from Lemma 6.11 where  $\psi = 0.9$ ,  $\psi' = (1 + 1/M)/1.14$ . Now notice that

$$\frac{|\mathcal{T}| + |\mathcal{R}|}{k} = \frac{0.9 - \frac{1+1/M}{1.14}}{1 - \frac{1+1/M}{1.14}} + (0.9 - 1/M) > 1,$$

for  $M \geq 86$ , therefore there exists some  $j \in \mathcal{R} \cap \mathcal{T}$ . For this  $j$ ,

$$\begin{aligned} (1 + 1/M)/1.14 \cdot (1.14\|\tilde{\mu} - \mu\|_2) &\leq B_{j,j}1.14\|\tilde{\mu} - \mu\|_2 \\ &\leq \langle Z_j - x, W_j \rangle \\ &= \langle Z_j - \mu, W_j \rangle + \langle \mu - \tilde{\mu}, W_j \rangle \\ &< B_{j,j}5Mr^* + \|\tilde{\mu} - \mu\|_2. \end{aligned}$$

Where the last step comes from  $\|W_j\|_2 \leq 1$ . This is easy to see in the pseudo-expectation language,

$$\begin{aligned} \|\tilde{\mathbf{E}}[b_j v]\|_2^2 &= \sum_{\ell} \tilde{\mathbf{E}}[b_j v_{\ell}]^2 \\ &\leq \sum_{\ell} \tilde{\mathbf{E}}[b_j^2] \tilde{\mathbf{E}}[v_{\ell}^2] && \text{(Cauchy-Schwarz)} \\ &\leq \tilde{\mathbf{E}}[\|v\|_2^2] && (\tilde{\mathbf{E}}[b_j^2] \leq 1) \\ &= 1. \end{aligned}$$

By rearranging the above inequality we obtain

$$B_{j,j} > 1/M \cdot \|\tilde{\mu} - \mu\|_2 (5Mr^*)^{-1} > 1,$$

which is a contradiction. Therefore,

$$\text{SDP}(\tilde{\mu}, 1.14\|\tilde{\mu} - \mu\|_2, Z) \leq 0.9k < 0.91k \leq \text{SDP}(\tilde{\mu}, \rho_s, Z).$$

Now by monotonicity of SDP in the second parameter, we conclude that  $\rho_s \leq 1.14\|\tilde{\mu} - \mu\|_2$ . Therefore

$$\hat{r} \leq \rho_s + r^* \leq 1.14\|\tilde{\mu} - \mu\|_2 + r^* \leq 1.15\|\tilde{\mu} - \mu\|_2,$$

as desired, where the final inequality comes from  $\|\tilde{\mu} - \mu\|_2 \geq 100(5 \times 100 + 1)r^* \geq 100r^*$ .  $\square$

**Corollary 6.19** (existence of  $r$  with SDP less than  $0.9k$ ). *In the proof of Lemma 6.18, we have proven that under Assumption 6.9, and the assumption that  $M \geq 86$ ,*

$$\text{SDP}(\tilde{\mu}, 1.14\|\mu - \tilde{\mu}\|_2, Z) \leq 0.9k.$$

## 6.5 Gradient Estimation

Suppose we have a current estimate  $\tilde{\mu}$  and  $\mu$  is a central point. Moreover,  $\tilde{\mu}$  and  $\mu$  are sufficiently far from each other. Our goal is to privately find a direction  $y$  such that  $\langle y, (\mu - \tilde{\mu}) / \|\mu - \tilde{\mu}\|_2 \rangle \geq \Omega(1)$ . In order to do this we run the exponential mechanism over a ball of radius  $1 - \Theta(1)$  in  $\mathbb{R}^d$  with score function SDP-VAL. We prove that SDP-VAL has low sensitivity (Section 6.5.5), is Lipschitz (Section 6.5.6), and is concave (Section 6.5.7). These properties ensure that we can sample from the corresponding distribution efficiently and that our algorithm is private. In order to prove the accuracy of the exponential mechanism (Section 6.5.9), we prove that there exists a large volume of points that have high scores (Section 6.5.8). If the exponential mechanism is accurate, it outputs a direction  $y$  from the ball that has high a high score. In Section 6.5.3, we show that if  $y$  has a high score it satisfies  $|\langle y, (\mu - \tilde{\mu}) / \|\mu - \tilde{\mu}\|_2 \rangle| \geq \Omega(1)$ . We wrap up the guarantees of the exponential mechanism with SDP-VAL in Theorem 6.20.

Now in order to decide between  $y$  and  $-y$ , we run the exponential mechanism over these two candidates with score function being the number  $Z_i - \tilde{\mu}$ 's that have positive inner product with each candidate. We show that at least one of the two candidates must have a high score and therefore with high probability it is outputted. Moreover the higher scoring candidate satisfies  $\langle y, (\mu - \tilde{\mu}) / \|\mu - \tilde{\mu}\|_2 \rangle \geq \Omega(1)$ . See Theorem 6.21 for the theorem statement, and Section 6.5.11 for its proof.

Finally, we put together the guarantees of Theorem 6.20, and Theorem 6.21 to obtain Theorem 6.22.

### 6.5.1 Algorithm

---

#### Algorithm 6 GRADIENT-ESTIMATION

---

**Input:** Parameters: Data points  $Z_1, \dots, Z_k \in \mathbb{R}^d$ , current estimate  $\tilde{\mu} \in \mathbb{R}^d$ , estimate of the distance  $d$

**Operation:**

1. Let  $M = 1000$ ,  $r = d/1.2$ ,  $\zeta = 1/(2M)$ .
2. Run the exponential mechanism over the ball of radius  $1 - \zeta$  with score function  $\text{SCORE}(y) = \text{SDP-VAL}(y; \tilde{\mu}, r, Z)$  to obtain  $y_0$ .
3. Run the exponential mechanism with score function  $\text{SCORE}(y) = \sum_{i=1}^k \mathbb{1}[\langle Z_i - \tilde{\mu}, y \rangle > 0]$  over  $\{y_0, -y_0\}$  to obtain  $\hat{y}$ .

**Output:**  $\hat{y}$

---

### 6.5.2 Main Theorems

The following theorem guarantees that the exponential mechanism in the second line of Algorithm 6, is private and can be run efficiently and outputs a direction  $y_0$  such that  $|\langle y_0, \Delta \rangle| \geq \Omega 1$  with high probability.

**Theorem 6.20** (SDP-VAL exponential mechanism). *Let  $Z_1 \dots Z_k, \tilde{\mu}, \mu \in \mathbb{R}^d$ ,  $r^* = 2\sqrt{k/n}$ ,  $M \geq 1000$ ,  $d \geq 0$ ,  $r = d/1.2$ ,  $\Delta = (\mu - \tilde{\mu}) / \|\mu - \tilde{\mu}\|_2$ . Suppose Assumption 6.9 holds,  $1.15\|\mu - \tilde{\mu}\|_2 \geq d \geq 0.9\|\mu - \tilde{\mu}\|_2$ ,  $\|\mu - \tilde{\mu}\|_2 \geq M(5Mr^*)$ ,  $\zeta = 1/(2M)$ . Then there exists a constant  $C'_M$  depending only on  $M$  such that the exponential mechanism in Algorithm 6 with score function*

$f(y) = \text{SDP-VAL}(y; \tilde{\mu}, r, Z)$  and privacy budget  $\varepsilon/T$ , over the ball of radius  $1 - \zeta$  in  $\mathbb{R}^d$ , returns  $y_0$  such that if  $k \geq C'_M T(d + \log(T/\beta))/\varepsilon$ , then

$$|\langle y_0, \Delta \rangle| \geq 0.1,$$

with probability  $1 - T/\beta$ . Moreover there exists an algorithm taking time  $\text{poly}(n, d, 1/\varepsilon)$  for sampling from the exponential mechanism.

*Proof.* Combination of Lemma 6.25 (sensitivity, privacy), Theorem 6.35 (accuracy of the exponential mechanism), Theorem 6.23 (accuracy guarantee of SDP-VAL), and Lemma 6.36 (runtime of the exponential mechanism). See Section 6.5.3 to Section 6.5.10 for the full proof.  $\square$

From the previous theorem we know that with high probability  $|\langle y_0, \tilde{\mu} - \mu \rangle| \Omega(1)$  The following theorem guarantees that the third line of the algorithm is private and that it has accuracy. That is, conditioned on the success of the previous step, with high probability it outputs a vector  $\hat{y}$  from  $\{y_0, -y_0\}$  such that  $\langle \hat{y}, (\mu - \tilde{\mu})/\|\mu - \tilde{\mu}\|_2 \rangle \geq \Omega(1)$ .

**Theorem 6.21** (exponential mechanism for the two directions). *Let  $Z_1, \dots, Z_k, \tilde{\mu}, \mu, y_0 \in \mathbb{R}^d$ ,  $r^* = 2\sqrt{k/n}$ ,  $\|\mu - \tilde{\mu}\|_2 \geq M(5Mr^*)$ ,  $\Delta = (\mu - \tilde{\mu})/\|\mu - \tilde{\mu}\|_2$  and  $M \geq 1000$ . Suppose  $|\langle y_0, \Delta \rangle| \geq 0.1$ , and that Assumption 6.9 holds. Then the exponential mechanism with score function  $\text{SCORE}(y) = \sum_{i=1}^k \mathbb{1}[\langle Z_i - \tilde{\mu}, y \rangle > 0]$ , over  $\{y_0, -y_0\}$ , and privacy budget  $\varepsilon/T$ , as in Algorithm 6, returns  $\hat{y}$  such that if  $k \geq 2MT(\log 2 + \log T/\beta)/\varepsilon$ , then*

$$\langle \hat{y}, \Delta \rangle \geq 0.1,$$

with probability  $1 - \beta/T$ .

*Proof.* See Section 6.5.11.  $\square$

Putting together Theorem 6.20, and Theorem 6.21 gives us the following algorithm, which guarantees that Algorithm 6 is private, has high utility with high probability, and that it can be run efficiently.

**Theorem 6.22** (GRADIENT-ESTIMATION). *Let  $Z_1, \dots, Z_k, \tilde{\mu}, \mu \in \mathbb{R}^d$ ,  $d \geq 0$ ,  $r^* = 2\sqrt{k/n}$ ,  $M \geq 1000$ ,  $d \geq 0$ ,  $\Delta = (\mu - \tilde{\mu})/\|\mu - \tilde{\mu}\|_2$ . Suppose Assumption 6.9 holds. Then there exists a constant  $C''_M$  depending only on  $M$  and  $\varepsilon/T$ -dp algorithm GRADIENT-ESTIMATION that if  $0.99\|\mu - \tilde{\mu}\|_2 \leq d \leq 1.15\|\mu - \tilde{\mu}\|_2$ ,  $\|\mu - \tilde{\mu}\|_2 \geq M(5Mr^*)$ , and  $k \geq C''_M T(d + \log T/\beta)/\varepsilon$ , returns  $\hat{y}$  such that*

$$\langle \hat{y}, \Delta \rangle > 0.1,$$

with probability  $1 - \beta/T$ . Moreover this algorithm has runtime  $\text{poly}(n, d, 1/\varepsilon)$ . We say that GRADIENT-ESTIMATION has been successful if the above guarantee holds.

In the rest of this section we prove Theorem 6.20, and Theorem 6.21.

### 6.5.3 SoS Analysis

**Theorem 6.23** (accuracy guarantee of SDP-VAL). *Let  $Z_1 \dots Z_k, \tilde{\mu}, \mu \in \mathbb{R}^d$ ,  $r^* = 2\sqrt{k/n}$ , and  $\Delta = (\mu - \tilde{\mu})/\|\mu - \tilde{\mu}\|_2$ . Suppose Assumption 6.9 holds and  $M \geq 1000$ ,  $d \geq 0.99\|\mu - \tilde{\mu}\|_2$ , and  $\|\mu - \tilde{\mu}\|_2 \geq M(5Mr^*)$ . Moreover, suppose  $\text{SDP-VAL}(y; \tilde{\mu}, d/1.2, Z) = \psi k$ , where  $\psi \geq 0.9$ . Then*

$$|\langle y, \Delta \rangle| \geq 0.1.$$

*Proof.* Let  $r = d/1.2$ , note that since  $d \geq 0.99\|\mu - \tilde{\mu}\|_2$ ,  $r \geq 0.8\|\mu - \tilde{\mu}\|_2$ . Suppose  $\tilde{\mathbf{E}}$  is the solution to SDP-VAL( $y; \tilde{\mu}, r, Z$ ) as in Definition 6.7. Therefore,

$$\begin{aligned} y &= \tilde{\mathbf{E}}[v] \\ \tilde{\mathbf{E}}[b_i^2] &= \tilde{\mathbf{E}}[b_i] \end{aligned} \quad (5)$$

$$\sum_i \tilde{\mathbf{E}}[v_i^2] \leq 1 \quad (6)$$

$$\tilde{\mathbf{E}}[b_i^2] \cdot r \leq \langle Z_i - \tilde{\mu}, \tilde{\mathbf{E}}[b_i v] \rangle.$$

Moreover,  $y$  has score  $\psi k$ , i.e.

$$\sum_i \tilde{\mathbf{E}}[b_i^2] = \psi k. \quad (7)$$

Let  $\mathcal{T}$  be the set that its existence is guaranteed from Lemma 6.11. Then  $|\mathcal{T}| \geq (\psi - 1/M)k$ , and

$$\forall i \in \mathcal{T} : \tilde{\mathbf{E}}[\langle Z_i - \mu, b_i v \rangle] < 5Mr^* \tilde{\mathbf{E}}[b_i^2]. \quad (8)$$

Also note that

$$(2\psi - 1 - 1/M)k \leq \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i^2] \leq \psi k. \quad (9)$$

Therefore we may write

$$\begin{aligned} r\psi k &= r \cdot \sum_i \tilde{\mathbf{E}}[b_i^2] \\ &= r \cdot \left[ \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i^2] + \sum_{i \notin \mathcal{T}} \tilde{\mathbf{E}}[b_i^2] \right] \\ &\leq \sum_{i \in \mathcal{T}} \langle Z_i - \tilde{\mu}, \tilde{\mathbf{E}}[b_i v] \rangle + (1 - \psi + 1/M)kr \\ &= \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle Z_i - \tilde{\mu}, v \rangle] + (1 - \psi + 1/M)kr \\ &= \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle Z_i - \mu, v \rangle] + \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle \mu - \tilde{\mu}, v - \tilde{\mathbf{E}}[v] \rangle] + \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle \mu - \tilde{\mu}, \tilde{\mathbf{E}}[v] \rangle] + (1 - \psi + 1/M)kr. \end{aligned}$$

Now we need to bound each term separately.

$$1. \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle Z_i - \mu, v \rangle].$$

$$\begin{aligned} \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle Z_i - \mu, v \rangle] &\leq 5Mr^* \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i^2] && \text{(Equation (8))} \\ &\leq 5Mkr^* \end{aligned}$$

$$2. \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle \mu - \tilde{\mu}, v - \tilde{\mathbf{E}}[v] \rangle].$$

$$\begin{aligned} \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle \mu - \tilde{\mu}, v - \tilde{\mathbf{E}}[v] \rangle] &\leq \langle \mu - \tilde{\mu}, \tilde{\mathbf{E}}[\sum_{i \in \mathcal{T}} b_i (v - \tilde{\mathbf{E}}[v])] \rangle \\ &\leq \langle \mu - \tilde{\mu}, \tilde{\mathbf{E}}[\sum_{i \in \mathcal{T}} (b_i - \tilde{\mathbf{E}}[b_i]) (v - \tilde{\mathbf{E}}[v])] \rangle \\ &\leq \|\mu - \tilde{\mu}\|_2 \cdot \|\tilde{\mathbf{E}}[\sum_{i \in \mathcal{T}} (b_i - \tilde{\mathbf{E}}[b_i]) (v - \tilde{\mathbf{E}}[v])]\|_2 \\ &\leq \|\mu - \tilde{\mu}\|_2 \cdot \tilde{\mathbf{E}}[\|\sum_{i \in \mathcal{T}} (b_i - \tilde{\mathbf{E}}[b_i]) (v - \tilde{\mathbf{E}}[v])\|_2] \quad (\text{Jensen's inequality}) \\ &= \|\mu - \tilde{\mu}\|_2 \cdot \tilde{\mathbf{E}}[\|\sum_{i \in \mathcal{T}} (b_i - \tilde{\mathbf{E}}[b_i])\| \cdot \|(v - \tilde{\mathbf{E}}[v])\|_2] \\ &\leq \|\mu - \tilde{\mu}\|_2 \cdot \tilde{\mathbf{E}}[(\sum_{i \in \mathcal{T}} (b_i - \tilde{\mathbf{E}}[b_i])^2)^{1/2}] \cdot \tilde{\mathbf{E}}[\|v - \tilde{\mathbf{E}}[v]\|_2^2]^{1/2} \end{aligned}$$

(Lemma B.4)

Now we need to bound the two last terms.

$$\begin{aligned} \tilde{\mathbf{E}} \left[ \left( \sum_{i \in \mathcal{T}} (b_i - \tilde{\mathbf{E}}[b_i]) \right)^2 \right] &= \tilde{\mathbf{E}} \left[ \left( \sum_{i \in \mathcal{T}} b_i - \tilde{\mathbf{E}}[\sum_{i \in \mathcal{T}} b_i] \right)^2 \right] \\ &= \tilde{\mathbf{E}}[(\sum_{i \in \mathcal{T}} b_i)^2] - \tilde{\mathbf{E}}[\sum_{i \in \mathcal{T}} b_i]^2 \quad (\text{Lemma B.5}) \\ &\leq \tilde{\mathbf{E}}[k \cdot \sum_{i \in \mathcal{T}} b_i^2] - \left( \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i] \right)^2 \quad (\text{Cauchy-Schwarz}) \\ &= k \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i^2] - \left( \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i^2] \right)^2 \quad (\text{Equation (5)}) \\ &\leq \psi k^2 - (2\psi - 1 - 1/M)^2 k^2 \quad (\text{Equation (9)}) \\ &= k^2 (\psi - (2\psi - 1 - 1/M)^2) \end{aligned}$$

$$\begin{aligned} \tilde{\mathbf{E}}[\|v - \tilde{\mathbf{E}}[v]\|_2^2] &= \tilde{\mathbf{E}}[\|v\|_2^2] - \|\tilde{\mathbf{E}}[v]\|_2^2 \quad (\text{Lemma B.5}) \\ &\leq \tilde{\mathbf{E}}[\|v\|_2^2] \\ &\leq 1 \quad (\text{Equation (6)}) \end{aligned}$$

Therefore,

$$\sum_i \tilde{\mathbf{E}}[b_i \langle \mu - \tilde{\mu}, v - \tilde{\mathbf{E}}[v] \rangle] \leq k \sqrt{\psi - (2\psi - 1 - 1/M)^2} \cdot \|\mu - \tilde{\mu}\|_2.$$

$$3. \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i] \langle \mu - \tilde{\mu}, \tilde{\mathbf{E}}[v] \rangle.$$

$$\sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i] = \sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i^2] \quad (\text{Equation (5)})$$

$$\leq \psi k \quad (\text{Equation (9)})$$

Therefore,

$$\sum_{i \in \mathcal{T}} \tilde{\mathbf{E}}[b_i \langle \mu - \tilde{\mu}, \tilde{\mathbf{E}}[v] \rangle] \leq \psi k |\langle \mu - \tilde{\mu}, \tilde{\mathbf{E}}[v] \rangle|.$$

Putting all parts together we obtain

$$\begin{aligned} (2\psi - 1 - 1/M)rk &\leq \sum_i \tilde{\mathbf{E}}[b_i \langle Z_i - \mu, v \rangle] + \sum_i \tilde{\mathbf{E}}[b_i \langle \mu - \tilde{\mu}, v - \tilde{\mathbf{E}}[v] \rangle] + \sum_i \tilde{\mathbf{E}}[b_i \langle \mu - \tilde{\mu}, \tilde{\mathbf{E}}[v] \rangle] \\ &\leq 5Mkr^* + k\sqrt{\psi - (2\psi - 1 - 1/M)^2} \cdot \|\mu - \tilde{\mu}\|_2 + \psi k |\langle \mu - \tilde{\mu}, \tilde{\mathbf{E}}[v] \rangle|. \end{aligned}$$

Therefore if  $\Delta = (\mu - \tilde{\mu})/\|\mu - \tilde{\mu}\|_2$ ,  $\tilde{d} = \|\mu - \tilde{\mu}\|_2$ .

$$\frac{1}{\psi} \cdot [(2\psi - 1 - 1/M) \cdot \frac{r}{\tilde{d}} - \frac{5Mr^*}{\tilde{d}} - \sqrt{\psi - (2\psi - 1 - 1/M)^2}] \leq |\langle \Delta, y \rangle|.$$

Now since  $0.9 \leq \psi \leq 1$ ,  $M \geq 1000$ ,  $r \geq 0.8\tilde{d}$ ,  $\tilde{d} \geq M(5Mr^*)$ , we get that

$$\begin{aligned} |\langle \Delta, y \rangle| &\geq 0.65(2\psi - 1 - 1/M) - \sqrt{\psi - (2\psi - 1 - 1/M)^2} + 0.15(2\psi - 1 - 1/M) - \frac{5Mr^*}{\tilde{d}} \\ &\geq 0.15(2 \times 0.9 - 1 - 1/1000) - 1/1000 \\ &> 0.1. \end{aligned}$$

□

The following corollary is useful in Section 6.3.

**Corollary 6.24** (SDP far). *Let  $Z_1, \dots, Z_k, \tilde{\mu}, \mu \in \mathbb{R}^d$ ,  $r^* = 2\sqrt{k/n}$ . Suppose Assumption 6.9 holds and  $M \geq 1000$ . Suppose  $\text{SDP}(\tilde{\mu}, r, Z) \geq 0.9k$ . Then*

$$\|\mu - \tilde{\mu}\|_2 \geq \frac{0.7r - 5Mr^*}{2}.$$

*Proof.* From the proof of Theorem 6.23, we have that

$$2\|\mu - \tilde{\mu}\|_2 + 5kr^* \geq rk(2 \times 0.9 - 1 - 1/M).$$

Rearranging the terms and noting that  $M \leq 1000$ , completes the proof. □

#### 6.5.4 Analysis of the SDP-VAL Exponential Mechanism

In order to analyze the exponential mechanism we need to analyze the following properties of SDP-VAL.

1. Sensitivity: for privacy analysis of the exponential mechanism.
2. Lipschitzness: for runtime analysis of the exponential mechanism.
3. Concavity: for runtime analysis of the exponential mechanism.
4. Volume of the set of *optimal* points: for accuracy analysis of the exponential mechanism.

In the following subsections we analyze these properties.



### 6.5.5 Sensitivity

**Lemma 6.25** (SDP-VAL sensitivity).  $\Delta_{\text{SDP-VAL}} \leq 1$ .

*Proof.* We want to prove that for every two neighboring datasets  $Z$  and  $Z'$

$$\max_{y, \tilde{\mu}, r} \max_{Z, Z'} |\text{SDP-VAL}(y; \tilde{\mu}, r, Z) - \text{SDP-VAL}(y; \tilde{\mu}, r, Z')| \leq 1.$$

If we prove that for every  $y, \tilde{\mu}, r$  and for every neighboring datasets  $Z, Z'$

$$\text{SDP-VAL}(y; \tilde{\mu}, r, Z) - 1 \leq \text{SDP-VAL}(y; \tilde{\mu}, r, Z'),$$

by symmetry, our desired statement will be implied. Suppose that the optimal solution for  $(y; \tilde{\mu}, r, Z)$  is  $(1, b, B, v, V, W)$ . Without loss of generality, suppose that  $Z$  and  $Z'$  differ in the first index. Now consider  $(1, b', B', v, V, W')$ , where it is equal to  $(1, b, B, v, V, W)$  everywhere, except at  $b_1, B_{1,\cdot}, B_{\cdot,1}, W_{1,\cdot}$ , where it is equal to zero, or in other words, in the second column and row of the PSD matrix. Then  $(1, b', B', v, V, W')$  will still satisfy the linear inequalities. Moreover the corresponding matrix for  $(1, b', B', v, V, W')$ , will be positive semidefinite, by Lemma B.1. Now note that  $\text{Tr } B' \geq \text{Tr } B - 1$ , and therefore,

$$\text{SDP-VAL}(y; \tilde{\mu}, r, Z) - 1 \leq \text{SDP-VAL}(y; \tilde{\mu}, r, Z'),$$

which completes the proof.  $\square$

The following corollary is useful in the analysis in Section 6.4.

**Corollary 6.26** (SDP sensitivity).  $\Delta_{\text{SDP}} \leq 1$ .

*Proof.* Same as Lemma 6.25.  $\square$

### 6.5.6 Lipschitzness

**Lemma 6.27.** *Let  $Z_1, \dots, Z_k, \tilde{\mu} \in \mathbb{R}^d, r \in \mathbb{R}, w > 0$ . Then there exists  $\alpha \in \mathbb{R}^d, \beta \in \mathbb{R}^k, \gamma \in \mathbb{R}; \lambda \in \mathbb{R}_{\geq 0}^k$ , and a degree-2 SoS polynomial  $\sigma \in \mathbb{R}[b_1 \dots, b_k, v_1, \dots, v_k]_{\leq 2}$  such that the following polynomial identity holds in variables  $b_1, \dots, b_k, v_1, \dots, v_d$ .*

$$\begin{aligned} \text{SDP-VAL}(y; \tilde{\mu}, r, Z) + \omega - \sum_{i=1}^k b_i^2 &= \sum_{i=1}^d \alpha_i (y_i - v_i) + \sum_{i=1}^k \beta_i (b_i - b_i^2) + \gamma (1 - \|v\|_2^2) \\ &\quad + \sum_{i=1}^k \lambda_i (\langle Z_i - \tilde{\mu}, bv \rangle - b_i^2 r) + \sigma(b, v) \end{aligned}$$

This lemma is a direct result of duality of pseudo-distributions and sos proofs (e.g. see [BS16]). Note that the constraints on  $b$  and  $v$  in the optimization problem imply boundedness (Lemma A.4) and therefore our problem satisfies the duality conditions.

Now we use this lemma to prove the following theorem.

**Lemma 6.28** (SDP-VAL Lipschitzness).  *$f(y) := \text{SDP-VAL}(y; \tilde{\mu}, r, Z)$  is  $L = \sqrt{dk}/\zeta$ -Lipschitz over the ball of radius  $(1 - \zeta)$  in  $\mathbb{R}^d$ , where  $0 < \zeta \leq 1$ .*

*Proof.* Let  $\omega > 0$ , then by Lemma 6.27, there exists  $\alpha \in \mathbb{R}^d; \beta \in \mathbb{R}^k; \gamma \in \mathbb{R}; \lambda \in \mathbb{R}_{\geq 0}^k$  and a degree-2 SoS polynomial  $\sigma$  such that

$$\begin{aligned} \text{SDP-VAL}(y; \tilde{\mu}, r, Z) + \omega - \sum_{i=1}^k b_i^2 &= \sum_{i=1}^d \alpha_i (y_i - v_i) + \sum_{i=1}^k \beta_i (b_i - b_i^2) + \gamma (1 - \|v\|_2^2) \\ &\quad + \sum_{i=1}^k \lambda_i (\langle Z_i - \tilde{\mu}, bv \rangle - b_i^2 r) + \sigma(b, v). \end{aligned}$$

First we prove that  $\alpha_i$ s are bounded. Note that the above equality, is a polynomial identity, therefore we may pick different values for  $b$  and  $v$ , and the equality should still hold. For  $1 \leq j \leq k$  let  $b = 0, v = y + te_j$  where  $t$  is chosen such that  $\|y + te_j\|_2 = 1$ . Such  $t$  exists because  $\|y\|_2 < 1 - \zeta$ . Moreover  $t > \zeta$  due to the triangle inequality. Now if we plug  $b$  and  $v$  into the above equality we obtain

$$\begin{aligned} \text{SDP-VAL}(y; \tilde{\mu}, r, Z) + \omega &= -\alpha_j t + \sigma(0, y + te_j) & (\sigma \geq 0, \gamma_i \geq 0) \\ &\geq -\alpha_j t. \end{aligned}$$

Therefore, for every  $1 \leq j \leq k$ ,

$$\alpha_j \geq -\frac{k + \omega}{\zeta}.$$

Similarly by letting  $b = 0, v = y - e_j t$ , we may obtain  $\alpha_j \leq (k + \omega)/\zeta$ , therefore  $|\alpha_j| \leq (k + \omega)/\zeta$ , and

$$\|\alpha\|_2 \leq \sqrt{d}(k + \omega)/\zeta.$$

Now that we have bounded  $\alpha$ , we can move on to proving Lipschitzness. Let

$$\begin{aligned} g(\alpha, \beta, \gamma, \lambda, \sigma; y; \tilde{\mu}, r, Z) &= \sum_{i=1}^k b_i^2 + \sum_{i=1}^d \alpha_i (y_i - v_i) + \sum_{i=1}^k \beta_i (b_i - b_i^2) + \gamma (1 - \|v\|_2^2) \\ &\quad + \sum_{i=1}^k \lambda_i (\langle Z_i - \tilde{\mu}, bv \rangle - b_i^2 r) + \sigma(b, v). \end{aligned}$$

Note that the co-domain of  $g$  is the set of degree-2 polynomials in  $(b, v)$ . Suppose  $y' = y + \delta u$ , where  $\|u\|_2 = 1$ . In order to prove Lipschitzness, we prove that there exists  $\alpha', \beta', \gamma', \lambda', \sigma'$  such that  $g(\alpha', \beta', \gamma', \lambda', \sigma'; y'; \tilde{\mu}, r, Z)$  is a degree-0 polynomial (i.e. it is a number in  $\mathbb{R}$ ), and is at most  $\text{SDP-VAL}(y; \tilde{\mu}, r, Z) + L\delta + h(\omega)$ , where  $\lim_{\omega \rightarrow 0} h(\omega) = 0$ . If we do so, by pseudo-expectation and SoS duality, we can conclude that for every  $\omega > 0$ ,  $\text{SDP-VAL}(y'; \tilde{\mu}, r, Z) < \text{SDP-VAL}(y; \tilde{\mu}, r, Z) + L\delta + h(\omega)$ . Taking  $\lim_{\omega \rightarrow 0}$  from both sides gives us  $\text{SDP-VAL}(y'; \tilde{\mu}, r, Z) \leq \text{SDP-VAL}(y; \tilde{\mu}, r, Z) + L\delta$ . By symmetry, this implies that  $|\text{SDP-VAL}(y'; \tilde{\mu}, r, Z) - \text{SDP-VAL}(y; \tilde{\mu}, r, Z)| \leq L\delta$  and that  $f$  is  $L$ -Lipschitz in  $y$ . It remains to construct a dual solution that satisfies the described properties. To do so, take  $(\alpha', \beta', \gamma', \lambda', \sigma') = (\alpha, \beta, \gamma, \lambda, \sigma)$ . Then

$$\begin{aligned} g(\alpha', \beta', \gamma', \lambda', \sigma'; y'; \tilde{\mu}, r, Z) &= g(\alpha, \beta, \gamma, \lambda, \sigma; y; \tilde{\mu}, r, Z) + \delta \sum_{i=1}^d \alpha_i u_i \\ &\leq \text{SDP-VAL}(y; \tilde{\mu}, r, Z) + \omega + \delta \|\alpha\|_2 \\ &\leq \text{SDP-VAL}(y; \tilde{\mu}, r, Z) + \omega + \delta \left( \sqrt{dk}/\zeta + \sqrt{d}\omega/\zeta \right) \\ &= \text{SDP-VAL}(y; \tilde{\mu}, r, Z) + \delta \sqrt{dk}/\zeta + \omega(\delta \sqrt{d}/\zeta + 1). \end{aligned}$$

as desired. Therefore  $f$  is  $L = \sqrt{dk}/\zeta$ -Lipschitz.  $\square$

### 6.5.7 Concavity

**Corollary 6.29** (SDP-VAL concavity). *Let  $Z_1, \dots, Z_k, \tilde{\mu} \in \mathbb{R}^d$ ,  $r \geq 0$ . Let  $f(y) := \text{SDP-VAL}(y; \tilde{\mu}, r, Z)$ . Then  $f$  is concave.*

*Proof.* Direct result of Lemma B.2. □

### 6.5.8 Volume

In this section we prove the following lemma.

**Lemma 6.30** (SDP-VAL volume ratio). *Let  $Z_1, \dots, Z_k, \tilde{\mu}, \mu \in \mathbb{R}^d$ ,  $r^* = 2\sqrt{k/n}$ ,  $1/(2M) \geq \zeta \geq 0$ . Suppose  $\|\tilde{\mu} - \mu\|_2 \geq M(5M)r^*$ ,  $M \geq 1000$ ,  $d \leq 1.15\|\mu - \tilde{\mu}\|_2$ , and that Assumption 6.9 holds. Then there exists a set  $\mathcal{H}^*$  inside  $\mathbb{B}_{1-\zeta}^d$ , the ball of radius  $1 - \zeta$  in  $\mathbb{R}^d$ , and a constant  $C_M$  depending only on  $M$  such that*

$$\log \frac{\text{vol}(\mathbb{B}_{1-\zeta}^d)}{\text{vol}(\mathcal{H}^*)} \leq C_M d,$$

and

$$\inf_{y \in \mathcal{H}^*} \text{SDP-VAL}(y; \tilde{\mu}, d/1.2, Z) \geq (1 - \frac{1}{M})k.$$

*Proof.* Putting together Lemma 6.31, Lemma 6.33, and Corollary 6.34, and setting  $\gamma = \theta = 1/M$ , and noting that  $d \leq 1.15\|\mu - \tilde{\mu}\|_2$  implies

$$\frac{d}{1.2} \leq 0.96\|\mu - \tilde{\mu}\|_2 \leq (1 - 1/M)(1 - 2/M)\|\mu - \tilde{\mu}\|_2.$$

Also note that from Corollary 6.34,  $C_M \geq \mathcal{O}(\log M)$  should suffice. □

The following lemma tells us that there's a set of good points. Later in this section we show that this set has a large volume inside the ball of radius  $1 - \zeta$ .

**Lemma 6.31** (goodness of the set). *Let  $Z_1, \dots, Z_k, \tilde{\mu} \in \mathbb{R}^d$ ,  $r^* = 2\sqrt{k/n}$ . Assume Assumption 6.9 holds. Let  $\tilde{d} = \|\mu - \tilde{\mu}\|_2$ ,  $\Delta = (\mu - \tilde{\mu})/\|\mu - \tilde{\mu}\|_2$ . Then for any  $0 \leq \gamma, \theta \leq 1$ , if  $\langle u, \Delta \rangle \geq 1 - \gamma$ ,  $\|u\|_2 = 1$ ,  $\tilde{d} \geq M(5M)r^*$ ,  $(1 - \theta)(1 - \gamma - 1/M)\tilde{d} \geq r$ ,*

$$\forall \alpha \in [1 - \theta, 1] : \text{SDP-VAL}(\alpha u; \tilde{\mu}, r, Z) \geq (1 - \frac{1}{M})k.$$

*Proof.* By Assumption 6.10 (which is a result of Assumption 6.9) we know that for the direction  $u$  there exists a set  $S_{-u}$  such that  $|S_{-u}| \geq k(1 - 1/M)$  and

$$\forall i \in S_{-u} : \langle Z_i - \mu, -u \rangle \leq 5Mr^*.$$

Now for every  $i \in S_{-u}$ , we have that

$$\begin{aligned} \langle Z_i - \tilde{\mu}, u \rangle &= \langle Z_i - \mu, u \rangle + \langle \mu - \tilde{\mu}, u \rangle \\ &\geq -5Mr^* + \|\mu - \tilde{\mu}\|_2 \langle u, \Delta \rangle \\ &\geq (1 - \gamma)\tilde{d} - 5Mr^* \\ &\geq (1 - \gamma - 1/M)\tilde{d}. \end{aligned}$$

Therefore for at least  $k(1 - 1/M)$  indices

$$\langle Z_i - \tilde{\mu}, \alpha u \rangle \geq (1 - \theta)(1 - \gamma - 1/M)\tilde{d} \geq r,$$

which completes the proof by Lemma A.5. □

The following corollary is useful in the analysis in Section 6.4.

**Corollary 6.32.** *Let  $Z_1, \dots, Z_k, \tilde{\mu} \in \mathbb{R}^d$ , and  $r^* = 2\sqrt{k/n}$ . Assume Assumption 6.9, holds,  $\|\mu - \tilde{\mu}\|_2 \geq M(5M)r^*$ . Then*

$$\text{SDP}(\tilde{\mu}, (1 - \frac{1}{M})\|\mu - \tilde{\mu}\|_2, Z) \geq (1 - \frac{1}{M})k.$$

The following lemma states that the set defined in Lemma 6.31, has a high volume.

**Lemma 6.33** (size of the good set). *Suppose  $\Delta \in \mathbb{R}^d$  such that  $\|\Delta\|_2 = 1$ ,  $0 \leq \gamma \leq 1$  and  $0 \leq \zeta \leq \theta \leq 1$ . Let*

$$S_{\theta, \zeta, \gamma, \Delta} = \left\{ \alpha u \mid \langle u, \Delta \rangle \geq 1 - \gamma, 1 - \theta \leq \alpha \leq 1 - \zeta, \|u\|_2 = 1, u \in \mathbb{R}^d \right\}.$$

Let  $\mathbb{B}_r^d$  denote the unit ball of radius  $r$  in  $\mathbb{R}^d$ . Then

$$\frac{\text{vol}(S_{\theta, \zeta, \gamma, \Delta})}{\text{vol}(\mathbb{B}_{1-\zeta}^d)} \geq \left( 1 - \left( \frac{1-\theta}{1-\zeta} \right)^d \right) \cdot \frac{1}{2} \left( \frac{\sqrt{1 - (1-\gamma)^2}}{2} \right)^{d-1}.$$

*Proof.* In order to prove this lemma we lower bound  $\text{vol}(S_{\theta, \zeta, \gamma, \Delta}) / \text{vol}(S_{1, \zeta, \gamma, \Delta})$ ,  $\text{vol}(S_{1, \theta, \gamma, \Delta}) / \text{vol}(S_{1, 0, \gamma, \Delta})$ ,  $\text{vol}(\mathbb{B}_1^d) / \text{vol}(\mathbb{B}_{1-\zeta}^d)$ .

Note that  $S_{1, \theta, \gamma, \Delta}$  is just a scaling of the set  $S_{1, \zeta, \gamma, \Delta}$  with respect to the origin with ratio  $(1 - \theta)/(1 - \zeta)$ . Therefore

$$\frac{\text{vol}(S_{1, \theta, \gamma, \Delta})}{\text{vol}(S_{1, \zeta, \gamma, \Delta})} = \left( \frac{1 - \theta}{1 - \zeta} \right)^d,$$

and

$$\frac{\text{vol}(S_{\theta, \zeta, \gamma, \Delta})}{\text{vol}(S_{1, \zeta, \gamma, \Delta})} = 1 - \left( \frac{1 - \theta}{1 - \zeta} \right)^d.$$

Similarly,

$$\frac{\text{vol}(S_{1, \zeta, \gamma, \Delta})}{\text{vol}(S_{1, 0, \gamma, \Delta})} = (1 - \zeta)^d.$$

To bound  $\text{vol}(S_{1, 0, \gamma, \Delta}) / \text{vol}(\mathbb{B}_1^d)$ , note that this ratio is equal to the ratio of the surface area of the corresponding cap of  $S_{1, 0, \gamma, \Delta}$  to the surface area of the unit sphere, therefore by Fact B.8,

$$\frac{\text{vol}(S_{1, 0, \gamma, \Delta})}{\text{vol}(\mathbb{B}_1^d)} \geq \frac{1}{2} \left( \frac{\sqrt{1 - (1 - \gamma)^2}}{2} \right)^{d-1}.$$

Now it remains to lower bound  $\text{vol}(\mathbb{B}_1^d) / \text{vol}(\mathbb{B}_{1-\zeta}^d)$ . Note that  $\mathbb{B}_1^d$  is a scaling of the set  $\mathbb{B}_{1-\zeta}^d$  with respect to the origin with ratio  $1/(1 - \zeta)$ , therefore,

$$\frac{\text{vol}(\mathbb{B}_1^d)}{\text{vol}(\mathbb{B}_{1-\zeta}^d)} = (1 - \zeta)^{-d}.$$

Multiplying the bounds gives us the desired inequality. □

**Corollary 6.34** (size of the good set). *Under the same assumptions as Lemma 6.33,*

$$\frac{\text{vol}(S_{\theta, \zeta, \gamma, \Delta})}{\text{vol}(\mathbb{B}_{1-\zeta}^d)} \geq \frac{\theta - \zeta}{2} \cdot (\gamma^{1/2}/2)^{d-1},$$

*More specifically, for every  $\theta, \zeta, \gamma$ , there exists some  $C_{\theta, \zeta, \gamma} \geq 0$ , such that*

$$\log \frac{\text{vol}(\mathbb{B}_{1-\zeta}^d)}{\text{vol}(S_{\theta, \zeta, \gamma, \Delta})} \leq C_{\theta, \zeta, \gamma} d.$$

### 6.5.9 Accuracy Analysis of the Exponential Mechanism for SDP-VAL

**Theorem 6.35** (accuracy guarantee of the exponential mechanism for SDP-VAL). *Let  $Z_1 \dots Z_k, \mu, \tilde{\mu} \in \mathbb{R}^d$ ,  $r^* = 2\sqrt{k/n}$ ,  $1/(2M) \geq \zeta \geq 0$ . Suppose  $M \geq 1000$ ,  $d \leq 1.15\|\mu - \tilde{\mu}\|_2$ ,  $r = d/1.2$ , and that Assumption 6.9 holds. Let  $\mathcal{M}_E$  be the exponential mechanism when instantiated with the score function  $f(y) = \text{SDP-VAL}(y; \tilde{\mu}, r, Z)$  and privacy budget  $\varepsilon/T$ , over the ball of radius  $1 - \zeta$  in  $\mathbb{R}^d$ , as in Algorithm 6. Let  $y_0$  denote the vector returned by the exponential mechanism, then there exists a constant  $C_M$  depending only on  $M$  such that*

$$\text{SDP-VAL}(y) \geq \left(1 - \frac{1}{M}\right)k - \frac{2T}{\varepsilon} (C_M d + \log(T/\beta)) ,$$

with probability  $1 - \beta/T$ . More specifically, if  $k \geq 2TM(C_M d + \log(T/\beta))/\varepsilon$ , then

$$\text{SDP-VAL}(y) \geq \left(1 - \frac{2}{M}\right)k ,$$

with probability  $1 - \beta/T$ .

*Proof.* This is a direct result of the volume based exponential mechanism, and our arguments about the volume of the good set and sensitivity i.e. Theorem 3.1, Lemma 6.30 and Lemma 6.25.  $\square$

### 6.5.10 Runtime Analysis of the Exponential Mechanism for SDP-VAL

**Lemma 6.36** (runtime of the exponential mechanism). *There exists a sampling scheme for the exponential mechanism for SDP-VAL in Algorithm 6 that takes time  $\text{poly}(n, d, 1/\varepsilon)$ .*

This lemma is a direct application of Theorem 3.2's sampling algorithm. Note that we have shown that SDP-VAL has Lipschitzness (Lemma 6.28), concavity (Corollary 6.29), and low sensitivity (Lemma 6.25), therefore we are allowed to use the mentioned result.

### 6.5.11 Proof of Theorem 6.21

First let's prove the following lemma.

**Lemma 6.37.** *Let  $Z_1 \dots, Z_k, \mu, \tilde{\mu}, y_0 \in \mathbb{R}^d$ ,  $\tilde{d} = \|\mu - \tilde{\mu}\|_2$ ,  $\Delta = |\mu - \tilde{\mu}|/\tilde{d}$ . Suppose  $|\langle y, \Delta \rangle| > 0.1$ ,  $\langle y, \Delta \rangle > 0$ ,  $\|\mu - \tilde{\mu}\|_2 \geq M(5Mr^*)$ ,  $M \geq 1000$ , and that Assumption 6.10 holds. Then*

$$|\{i \mid \langle Z_i - \tilde{\mu}, y \rangle > 0\}| \geq (1 - 1/M)k .$$

*Proof.* The proof is similar to [CFB19]. By Assumption 6.10, we know that there exists a set  $G$ , such that  $|G| \geq (1 - 1/M)k$  and

$$\forall i \in G : \langle Z_i - \mu, -y \rangle \leq 5Mr^* .$$

Therefore, for every  $i \in G$ ,

$$\begin{aligned} \langle Z_i - \tilde{\mu}, y \rangle &= \langle Z_i - \mu, y \rangle + \langle \mu - \tilde{\mu}, y \rangle \\ &\geq -5Mr^* + \langle \Delta, y \rangle \|\mu - \tilde{\mu}\|_2 \\ &\geq -5Mr^* + 0.1M(5Mr^*) \\ &> 0, \end{aligned}$$

which completes the proof.  $\square$

Now we use this lemma to prove Theorem 6.21.

*Proof of Theorem 6.21.* Since either  $\langle y_0, \Delta \rangle > 0$ , or  $\langle -y_0, \Delta \rangle > 0$ , by Lemma 6.37, and the guarantee of the exponential mechanism, the exponential mechanism returns  $\hat{y}$  such that

$$\text{SCORE}(\hat{y}) \geq (1 - 1/M)k - 2T/\varepsilon(\log 2 + \log T/\beta),$$

with probability  $1 - \beta/T$ . Since  $k \geq 2MT(\log 2 + \log T/\beta)/\varepsilon$ , the exponential mechanism returns  $\hat{y}$  such that

$$\text{SCORE}(\hat{y}) \geq (1 - 2/M)k,$$

with probability  $1 - \beta/T$ . If  $\text{SCORE}(\hat{y}) \geq (1 - 2/M)k$ , then by Lemma 6.37, and the assumption that  $M \geq 1000$ ,  $\text{SCORE}(\hat{y}) \geq (1 - 2/M)k$  implies that  $\langle \hat{y}, \Delta \rangle > 0$ , as desired.  $\square$

## 6.6 Gradient Descent

We show that if the algorithm has not stopped in a round of the algorithm and that every step of the algorithm i.e. HALT-ESTIMATION, DISTANCE-ESTIMATION, and GRADIENT-ESTIMATION, have been successful, then we move closer to  $\mu$  (the central point) linearly (Lemma 6.38).

Finally we conclude that if all of the steps of the algorithm are successful then either we halt and return a point with optimal distance or we get closer to the central point  $\mu$ , linearly (Theorem 6.39)

**Lemma 6.38** (Gradient Descent Step). *Let  $Z_1, \dots, Z_k, \tilde{\mu}_{t-1}, \mu \in \mathbb{R}^d$ ,  $\Delta_{t-1} = (\mu - \tilde{\mu}_{t-1})/\|\mu - \tilde{\mu}_{t-1}\|_2$ , the step-size  $\eta = 0.075$ . Suppose in the  $t$ -th round inside the for loop in Algorithm 3, we have not halted, and all of HALT-ESTIMATION, DISTANCE-ESTIMATION, and GRADIENT-ESTIMATION, have been successful, i.e. the guarantees in Theorem 6.13, Theorem 6.14, and Theorem 6.22 are satisfied and*

$$0.99\|\tilde{\mu}_{t-1} - \mu\|_2 \leq d_t \leq 1.15\|\tilde{\mu}_{t-1} - \mu\|_2, \quad \langle g_t, \Delta_{t-1} \rangle > 0.1, \quad \|g_t\|_2 \leq 1.$$

Then

$$\|\tilde{\mu}_t - \mu\|_2 \leq 0.999\|\tilde{\mu}_{t-1} - \mu\|_2.$$

*Proof.* The proof is similar to [CFB19]. Let  $\tilde{d}_{t-1} = \|\tilde{\mu}_{t-1} - \mu\|_2$ . Then

$$\begin{aligned} \|\tilde{\mu}_t - \mu\|_2^2 &= \|\tilde{\mu}_t - \tilde{\mu}_{t-1} + \tilde{\mu}_{t-1} - \mu\|_2^2 \\ &= \|g_t d_t \eta\|_2^2 + \tilde{d}_{t-1}^2 + 2\langle g_t d_t \eta, \tilde{\mu}_{t-1} - \mu \rangle \\ &\leq \tilde{d}_{t-1}^2(1 + 1.15^2 \eta^2) + 2\langle g_t d_t \eta, -\Delta_{t-1} \rangle \\ &\leq \tilde{d}_{t-1}^2(1 + 1.15^2 \eta^2 - 2 \times 0.99 \times 0.1 \times \eta) \\ &= 0.993\|\tilde{\mu}_{t-1} - \mu\|_2^2. \end{aligned}$$

Taking square root gives us the final result.  $\square$

**Theorem 6.39** (Gradient Descent). *Let  $Z_1, \dots, Z_k, \mu_0 \in \mathbb{R}^d$ ,  $M \geq 1000$ ,  $r^* = 2\sqrt{k/n}$ . Suppose the guarantees in Theorem 6.13, Theorem 6.14, and Theorem 6.22, have been successful in every round  $t$ . Then two outcomes are possible, either Algorithm 3 doesn't halt and returns  $\tilde{\mu}^*$  such that*

$$\|\tilde{\mu}^* - \mu\|_2 \leq (0.999)^T \|\mu_0 - \mu\|_2,$$

*or Algorithm 3, halts and returns  $\tilde{\mu}^*$  such that*

$$\|\tilde{\mu}^* - \mu\|_2 \leq (4M(5M + 1) + 15M)r^*.$$

*Proof.* Putting together Lemma 6.38, and Theorem 6.13.  $\square$

## 6.7 Robustness

**Remark 6.40** (Robustness of Fine Estimation). *Note that SDP, and SDP-VAL, and our quadratic programs all have sensitivity 1 by Lemma 6.25, and Corollary 6.26. There exists a universal constant  $C_0$  (say  $C_0 \leq 1/1000$ ), such that our arguments would still hold under  $C_0 k$ -fraction corruption of the bucketed means  $(Z_1, \dots, Z_k)$ . Now suppose that  $m = C_0/\eta_0$ , then under  $\eta \leq \eta_0$  corruption of the samples  $(X_1, \dots, X_n)$ , SDP, SDP-VAL, and our quadratic programs are perturbed by at most*

$$\eta n \leq \eta_0 m k \leq C_0 k.$$

*Therefore, the guarantees of the algorithm will still hold. Moreover, in this setting, by Theorem 6.12, we obtain an estimate  $\tilde{\mu}^*$ , where  $\|\mu - \tilde{\mu}^*\|_2 \leq \mathcal{O}(\frac{1}{\sqrt{m}}) = \mathcal{O}(\sqrt{\eta_0})$ , as desired.*

## 7 Lower Bounds

In this section, we prove lower bounds for mean estimation. While lower bounds for these settings were known previously [HT10, BD14, BKSU19, KSU20], in contrast to prior work, we focus explicitly on the failure probability  $\beta$ . We first establish a general purpose lower bound framework, and then apply it to heavy-tailed and Gaussian distributions. Technically, we follow the standard packing lower bound technique.

**Theorem 7.1.** *Let  $\mathcal{P} = \{P_1, \dots, P_m\}$  be a set of distributions, and  $P_O$  be a distribution such that for every  $P_i \in \mathcal{P}$ ,  $\|P_i - P_O\|_{\text{TV}} \leq \gamma$ . Let  $\mathcal{G} = \{G_1, \dots, G_m\}$  be a collection of disjoint subsets of some set  $\mathcal{Y}$ . If there is an  $\varepsilon$ -DP algorithm  $M$  such that  $\mathbb{P}_{X \sim P_i^n}[M(X) \in G_i] \geq \beta$  for all  $i \in [m]$ , then*

$$n \geq \Omega\left(\frac{\log m + \log(1/\beta)}{\gamma(e^{2\varepsilon} - 1)}\right).$$

Note that for the usual regime  $\varepsilon \leq 1$ , we can replace the  $e^{2\varepsilon} - 1$  in the denominator with  $\varepsilon$ .

*Proof.* Assume there exists some mechanism  $M$  that satisfies the theorem's assumptions. By the definition of  $\varepsilon$ -DP and group privacy, we know that for any two datasets  $D, D'$ , and any  $i \in [m]$  we have

$$\mathbb{P}[M(D) \in G_i] \leq e^{\varepsilon \# [D \neq D']} \mathbb{P}[M(D') \in G_i], \quad (10)$$

where  $\# [D \neq D']$  denotes the Hamming distance between  $D$  and  $D'$ . Note that for the time being,  $D$  and  $D'$  are fixed datasets (as opposed to random variables), and  $0 \leq \# [D \neq D'] \leq n$ .

Basic facts about coupling imply that, for any two distributions  $P_i, P_j \in \mathcal{P}$ , there exists a coupling  $\omega_{i,j}^*$  of  $P_i^n$  and  $P_j^n$  such that if  $(X_i, X_j) \sim \omega_{i,j}^*$ , we have

$$\# [X_i \neq X_j] \sim \text{Binomial}(n, \|P_i - P_j\|_{\text{TV}}). \quad (11)$$

Note that by the triangle inequality (through  $P_O$ ), we have that  $\|P_i - P_j\|_{\text{TV}} \leq 2\gamma$ .

By taking expected value of both sides of Equation (10), we get

$$\mathbf{E}_{(X_i, X_j) \sim \omega^*} [\mathbb{P}[M(X_j) \in G_j]] \leq \mathbf{E}_{(X_i, X_j) \sim \omega^*} \left[ e^{\varepsilon \# [X_i \neq X_j]} \cdot \mathbb{P}[M(X_i) \in G_j] \right], \quad (12)$$

for any two  $P_i, P_j$ .

For every  $t \in [m]$ , define  $Z_t^{(i)} = \mathbb{P}_M[M(X_i) \in G_t]$ . Note that the probability here is only over  $M$ ,  $Z_t^{(i)}$  remains a random variable which depends on the sampling  $X_i \sim P_i^n$ . Stated differently, to

evaluate the random variable  $Z_t^{(i)}$ , one first samples  $X_i \sim P_i^n$ , and then computes  $\mathbb{P}_M [M(X_i) \in G_t]$  for the realized dataset  $X_i$ . Since the  $G_t$ 's are disjoint, the law of total probability implies  $\sum_t Z_t^{(i)} \leq 1$  for any  $i \in [m]$ . By the accuracy guarantee of the mechanism  $M$ , we know that

$$\mathbf{E}_{X_i \sim P_i^n} [Z_t^{(i)}] = \mathbb{P}_{X_i \sim P_i^n, M} [M(X_i) \in G_t] \geq 1 - \beta,$$

therefore  $\mathbf{E}_{X_i \sim P_i^n} \left[ \sum_{t \neq i} Z_t^{(i)} \right] \leq \beta$ . Since  $Z_t^{(i)} \leq 1$ ,

$$\sum_{t \neq i} \mathbf{E} \left[ \left( Z_t^{(i)} \right)^2 \right] \leq \sum_{t \neq i} \mathbf{E} [Z_t^{(i)}] \leq \beta.$$

Therefore by the pigeonhole principle, for every  $i$ , there exists some  $\ell$  such that

$$\mathbf{E}_{X_i \sim P_i^n} \left[ \mathbb{P} [M(X_i) \in G_\ell]^2 \right] = \mathbf{E}_{X_i \sim P_i^n} \left[ \left( Z_\ell^{(i)} \right)^2 \right] \leq \frac{\beta}{m-1}. \quad (13)$$

Now we take  $j = \ell$  in Equation (12). We get

$$\mathbf{E}_{(X_i, X_\ell) \sim \omega^*} \left[ \mathbb{P} [M(X_\ell) \in G_\ell] \right] \leq \mathbf{E}_{(X_i, X_\ell) \sim \omega^*} \left[ e^{\varepsilon \# [X_i \neq X_\ell]} \cdot \mathbb{P} [M(X_i) \in G_\ell] \right].$$

By the accuracy guarantee of  $M$  we know that we can lower bound the left-hand side by  $1 - \beta$ . We upper bound the right-hand side by Hölder's inequality.

$$\mathbf{E}_{(X_i, X_\ell) \sim \omega^*} \left[ e^{\varepsilon \# [X_i \neq X_\ell]} \cdot \mathbb{P} [M(X_i) \in G_\ell] \right] \leq \mathbf{E}_{(X_i, X_\ell) \sim \omega^*} \left[ e^{2\varepsilon \# [X_i \neq X_\ell]} \right]^{1/2} \cdot \mathbf{E}_{(X_i, X_\ell) \sim \omega^*} \left[ \mathbb{P} [M(X_i) \in G_\ell]^2 \right]^{1/2}.$$

For the second term, Equation (13) implies that

$$\mathbf{E}_{(X_i, X_\ell) \sim \omega^*} \left[ \mathbb{P} [M(X_i) \in G_\ell]^2 \right] = \mathbf{E}_{X_i \sim P_i^n} \left[ \mathbb{P} [M(X_i) \in G_\ell]^2 \right] \leq \frac{\beta}{m-1}.$$

It remains to bound the first term. By Equation (11), we know that  $\# [X_i \neq X_\ell] \sim \text{Binomial}(n, \|P_i - P_\ell\|_{\text{TV}})$ , and thus, this term is just the moment-generating function of the Binomial distribution, evaluated at the point  $2\varepsilon$ . Since the MGF of  $\text{Binomial}(n, p)$  is  $(1 - p + pe^t)^n$ ,

$$\mathbf{E}_{(X_i, X_\ell) \sim \omega^*} \left[ e^{2\varepsilon \# [X_i \neq X_\ell]} \right]^{1/2} = (1 - p + pe^{2\varepsilon})^{n/2} \leq \exp \left( \frac{n}{2} \cdot \log(1 + 2\gamma(e^{2\varepsilon} - 1)) \right).$$

Note that  $x \geq \log(1 + x)$ , therefore

$$\mathbf{E}_{(X_i, X_\ell) \sim \omega^*} \left[ e^{2\varepsilon \# [X_i \neq X_\ell]} \right]^{1/2} \leq \exp(n\gamma \cdot (e^{2\varepsilon} - 1)).$$

Combining these derivations,

$$1 - \beta \leq \exp(n\gamma(e^{2\varepsilon} - 1)) \cdot \sqrt{\frac{\beta}{m-1}}.$$

Therefore,

$$n \geq \Omega \left( \frac{\log \left( \frac{(1-\beta)^2}{\beta} \right) + \log m}{\gamma(e^{2\varepsilon} - 1)} \right).$$

Since  $\beta < 1$ , we can conclude

$$n \geq \Omega \left( \frac{\log m + \log \frac{1}{\beta}}{\gamma(e^{2\varepsilon} - 1)} \right).$$

□



Now, we apply this theorem to derive lower bounds for mean estimation for distributions with bounded moments.

**Theorem 7.2.** *Suppose there exists an  $\varepsilon$ -DP algorithm  $M$  such that for every distribution  $D$  on  $\mathbb{R}^d$  such that  $\|\mathbf{E}_{X \sim D} X\| \leq R$  and  $\mathbf{E}_{X \sim D} [|\langle X - \mathbf{E}_{X \sim D}[X], v \rangle|^k] \leq 1$  for some  $k \geq 2$  and every unit vector  $v \in \mathbb{R}^d$ , given  $X_1, \dots, X_n \sim D$ , with probability  $1 - \beta$  the algorithm  $M$  outputs  $\hat{\mu}$  such that  $\|\hat{\mu} - \mathbf{E}_{X \sim D} X\| \leq \alpha$  where  $\alpha$  is smaller than some absolute constant. Then we have that*

$$n \geq \Omega \left( \frac{d + \log(1/\beta)}{\alpha^2} + \frac{d + \log(1/\beta)}{\alpha^{\frac{k}{k-1}} \varepsilon} + \frac{d \log R + \log(1/\beta)}{\varepsilon} \right).$$

*Proof.* The first term is folklore from the non-private setting and obtains even for estimating the mean of a Gaussian.

To obtain the last term, consider a maximum 1-packing of the ball with  $\ell_2$ -radius  $R$ . Standard results imply that this will be of size  $R^{\Omega(d)}$ . Consider a point mass distribution at each of these points. Each distribution clearly satisfies the mean and bounded moment condition in the theorem statement. We apply Theorem 7.1 with  $\mathcal{P}$  being this set of distributions, noting that  $\gamma \leq 1$  trivially. Combining the 1-packing property with our accuracy requirement that  $\alpha \leq 1/2$ , we get that the set of sets  $\mathcal{G}$  are indeed disjoint. This gives the third term.

To obtain the second term, we employ a construction of Barber and Duchi [BD14]. We consider a set of distributions parameterized by unit vectors  $v \in \mathbb{R}^d$ . The distribution parameterized by  $v$  has mass  $1 - 25\alpha^{\frac{k}{k-1}}$  at the origin and mass  $25\alpha^{\frac{k}{k-1}}$  on the point  $\frac{1}{6}\alpha^{-\frac{1}{k-1}}v$ .

We show that each of these distributions satisfies the conditions of the theorem statement. The mean of this distribution is  $25\alpha^{\frac{k}{k-1}}\frac{1}{6}\alpha^{-\frac{1}{k-1}}v = \frac{25}{6}\alpha v$ , which has  $\ell_2$  norm  $\leq R$  for  $R$  larger than an absolute constant. We check the bounded moment condition for an arbitrary unit vector  $u$ :

$$\begin{aligned} \mathbf{E}[|\langle X - \mathbf{E}[X], u \rangle|^k] &\leq \mathbf{E}[|\langle X - \mathbf{E}[X], v \rangle|^k] \\ &\leq (1 - 25\alpha^{\frac{k}{k-1}}) \left(\frac{25}{6}\alpha\right)^k + 25\alpha^{\frac{k}{k-1}} \left(\frac{1}{6}\alpha^{-\frac{1}{k-1}}\right)^k \\ &\leq 1. \end{aligned}$$

The last inequality uses the bounds on  $\alpha$  and  $k$ . Therefore, each such distribution satisfies the theorem conditions.

Now, we look to apply Theorem 7.1. Let  $\mathcal{P}$  be the set of distributions parameterized by a maximum set of points  $V$  on the unit sphere such that for each  $v, v' \in V$ , we have that  $\|v - v'\| \geq 1/2$ . Standard results imply that this set will be of size at least  $2^{\Omega(d)}$ . All distributions are within distance  $\gamma = 25\alpha^{\frac{k}{k-1}}$  from the point mass distribution on the origin. Finally, since for each pair  $v, v'$  we have  $\|v - v'\| \geq 1/2$ , this implies that the means of any two distributions are at distance  $\geq \frac{25}{12}\alpha$  from each other, and the balls of radius  $\alpha$  around each mean are disjoint. Letting these be the  $G_i$ 's, we conclude the theorem statement using Theorem 7.1.  $\square$

We can similarly derive a lower bound for mean estimation of Gaussian distributions.

**Theorem 7.3.** *Suppose there exists an  $\varepsilon$ -DP algorithm  $M$  such that for every Gaussian distribution  $\mathcal{N}(\mu, I)$  on  $\mathbb{R}^d$  such that  $\|\mu\| \leq R$ , given  $X_1, \dots, X_n \sim D$ , with probability  $1 - \beta$  the algorithm  $M$  outputs  $\hat{\mu}$  such that  $\|\hat{\mu} - \mathbf{E}_{X \sim D} X\| \leq \alpha$  where  $\alpha$  is smaller than some absolute constant. Then we have that*

$$n \geq \Omega \left( \frac{d + \log(1/\beta)}{\alpha^2} + \frac{d + \log(1/\beta)}{\alpha \varepsilon} + \frac{d \log R + \log(1/\beta)}{\varepsilon} \right).$$

*Proof.* The first term is again standard from the non-private literature, and the third term can be obtained akin to in the proof of Theorem 7.2 (though we consider a Gaussian with each center, rather than a point mass).

The second term is also similar to that of Theorem 7.2. The construction will involve the set of Gaussians with mean  $4\alpha v$  where  $v$  is a unit vector. These are Gaussian with bounded mean, thus satisfying the conditions of the theorem statement.

As before, we try to apply Theorem 7.1. We let  $\mathcal{P}$  be the set of  $2^{\Omega(d)}$  distributions associated with a packing of unit vectors  $V$  such that for each  $v, v' \in V$ , we have that  $\|v - v'\| \geq 1/2$ . Each pair of means is at distance  $\geq 2\alpha$ , and thus the balls of radius  $\alpha$  around each mean are disjoint. The proof is concluded by noting that all these distributions are at total variation distance  $\gamma = O(\alpha)$  from  $N(0, I)$ , due to the standard fact that  $\ell_2$  distance between means and total variation distance are equivalent (up to constant factors) for identity-covariance Gaussians when the means are at distance smaller than an absolute constant.  $\square$

## Acknowledgements

The authors would like to thank Adam Smith for his role in the conception and contributions in the early stages of this project.

## References

- [AAK21] Ishaq Aden-Ali, Hassan Ashtiani, and Gautam Kamath. On the sample complexity of privately learning unbounded high-dimensional gaussians. In *Proceedings of the 32nd International Conference on Algorithmic Learning Theory, ALT '21*, pages 185–216. JMLR, Inc., 2021.
- [AAL21] Ishaq Aden-Ali, Hassan Ashtiani, and Christopher Liaw. Privately learning mixtures of axis-aligned gaussians. In *Advances in Neural Information Processing Systems 34*, NeurIPS '21. Curran Associates, Inc., 2021.
- [ADK<sup>+</sup>19a] Kareem Amin, Travis Dick, Alex Kulesza, Andres Munoz, and Sergei Vassilvitskii. Differentially private covariance estimation. In *Advances in Neural Information Processing Systems 32*, NeurIPS '19, pages 14190–14199. Curran Associates, Inc., 2019.
- [ADK19b] Brendan Avent, Yatharth Dubey, and Aleksandra Korolova. The power of the hybrid model for mean estimation. *Proceedings on Privacy Enhancing Technologies*, 2020(4):48–68, 2019.
- [AL21] Hassan Ashtiani and Christopher Liaw. Private and polynomial time algorithms for learning Gaussians and beyond. *arXiv preprint arXiv:2111.11320*, 2021.
- [AMB19] Marco Avella-Medina and Victor-Emmanuel Brunel. Differentially private sub-Gaussian location estimators. *arXiv preprint arXiv:1906.11923*, 2019.
- [Bal97] Keith Ball. An elementary introduction to modern convex geometry. 1997.
- [BBDS12] Jeremiah Blocki, Avrim Blum, Anupam Datta, and Or Sheffet. The Johnson-Lindenstrauss transform itself preserves differential privacy. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS '12*, pages 410–419, Washington, DC, USA, 2012. IEEE Computer Society.

- [BD14] Rina Foygel Barber and John C Duchi. Privacy and statistical risk: Formalisms and minimax bounds. *arXiv preprint arXiv:1412.4451*, 2014.
- [BDKU20] Sourav Biswas, Yihe Dong, Gautam Kamath, and Jonathan Ullman. Coinpress: Practical private mean and covariance estimation. In *Advances in Neural Information Processing Systems 33*, NeurIPS '20, pages 14475–14485. Curran Associates, Inc., 2020.
- [BGS<sup>+</sup>21] Gavin Brown, Marco Gaboardi, Adam Smith, Jonathan Ullman, and Lydia Zakyntinou. Covariance-aware private mean estimation without private covariance estimation. *arXiv preprint arXiv:2106.13329*, 2021.
- [BKSW19] Mark Bun, Gautam Kamath, Thomas Steinke, and Zhiwei Steven Wu. Private hypothesis selection. In *Advances in Neural Information Processing Systems 32*, NeurIPS '19, pages 156–167. Curran Associates, Inc., 2019.
- [BS16] Boaz Barak and David Steurer. Proofs, beliefs, and algorithms through the lens of sum-of-squares. *Course notes: <http://www.sumofsquares.org/public/index.html>*, 2016.
- [BS19] Mark Bun and Thomas Steinke. Average-case averages: Private algorithms for smooth sensitivity and mean estimation. In *Advances in Neural Information Processing Systems 32*, NeurIPS '19, pages 181–191. Curran Associates, Inc., 2019.
- [BST14] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *Proceedings of the 55th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '14, pages 464–473, Washington, DC, USA, 2014. IEEE Computer Society.
- [BUV14] Mark Bun, Jonathan Ullman, and Salil Vadhan. Fingerprinting codes and the price of approximate differential privacy. In *Proceedings of the 46th Annual ACM Symposium on the Theory of Computing*, STOC '14, pages 1–10, New York, NY, USA, 2014. ACM.
- [CFB19] Yeshwanth Cherapanamjeri, Nicolas Flammarion, and Peter L. Bartlett. Fast mean estimation with sub-Gaussian rates. In *Proceedings of the 32nd Annual Conference on Learning Theory*, COLT '19, pages 786–806, 2019.
- [CWZ19] T. Tony Cai, Yichen Wang, and Linjun Zhang. The cost of privacy: Optimal rates of convergence for parameter estimation with differential privacy. *arXiv preprint arXiv:1902.04495*, 2019.
- [DFM<sup>+</sup>20] Wenxin Du, Canyon Foot, Monica Moniot, Andrew Bray, and Adam Groce. Differentially private confidence intervals. *arXiv preprint arXiv:2001.02285*, 2020.
- [DHS15] Ilias Diakonikolas, Moritz Hardt, and Ludwig Schmidt. Differentially private learning of structured discrete distributions. In *Advances in Neural Information Processing Systems 28*, NIPS '15, pages 2566–2574. Curran Associates, Inc., 2015.
- [DK19] Ilias Diakonikolas and Daniel M. Kane. Recent advances in algorithmic high-dimensional robust statistics. *arXiv preprint arXiv:1911.05911*, 2019.
- [DKK<sup>+</sup>16] Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *Proceedings of the 57th Annual IEEE Symposium on Foundations*

- of Computer Science*, FOCS '16, pages 655–664, Washington, DC, USA, 2016. IEEE Computer Society.
- [DKK<sup>+</sup>19] Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high-dimensions without the computational intractability. *SIAM Journal on Computing*, 48(2):742–864, 2019.
- [DKP20] Ilias Diakonikolas, Daniel M Kane, and Ankit Pensia. Outlier robust mean estimation with subgaussian rates via stability. In *Advances in Neural Information Processing Systems 33*, NeurIPS '20, pages 1830–1840. Curran Associates, Inc., 2020.
- [DL09] Cynthia Dwork and Jing Lei. Differential privacy and robust statistics. In *Proceedings of the 41st Annual ACM Symposium on the Theory of Computing*, STOC '09, pages 371–380, New York, NY, USA, 2009. ACM.
- [DL19] Jules Depersin and Guillaume Lecué. Robust subgaussian estimation of a mean vector in nearly linear time. *arXiv preprint arXiv:1906.03058*, 2019.
- [DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography*, TCC '06, pages 265–284, Berlin, Heidelberg, 2006. Springer.
- [DN03] Irit Dinur and Kobbi Nissim. Revealing information while preserving privacy. In *Proceedings of the 22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '03, pages 202–210, New York, NY, USA, 2003. ACM.
- [DSS<sup>+</sup>15] Cynthia Dwork, Adam Smith, Thomas Steinke, Jonathan Ullman, and Salil Vadhan. Robust traceability from trace amounts. In *Proceedings of the 56th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '15, pages 650–669, Washington, DC, USA, 2015. IEEE Computer Society.
- [DSSU17] Cynthia Dwork, Adam Smith, Thomas Steinke, and Jonathan Ullman. Exposed! a survey of attacks on private data. *Annual Review of Statistics and Its Application*, 4(1):61–84, 2017.
- [EKKL20] Marek Eliáš, Michael Kapralov, Janardhan Kulkarni, and Yin Tat Lee. Differentially private release of synthetic graphs. In *Proceedings of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '20, pages 560–578, Philadelphia, PA, USA, 2020. SIAM.
- [HKP<sup>+</sup>17] Samuel B Hopkins, Pravesh K Kothari, Aaron Potechin, Prasad Raghavendra, Tselil Schramm, and David Steurer. The power of sum-of-squares for detecting hidden structures. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 720–731. IEEE, 2017.
- [HL18] Samuel B. Hopkins and Jerry Li. Mixture models, robustness, and sum of squares proofs. In *Proceedings of the 50th Annual ACM Symposium on the Theory of Computing*, STOC '18, pages 1021–1034, New York, NY, USA, 2018. ACM.
- [HLY21] Ziyue Huang, Yuting Liang, and Ke Yi. Instance-optimal mean estimation under differential privacy. In *Advances in Neural Information Processing Systems 34*, NeurIPS '21. Curran Associates, Inc., 2021.

- [HLZ20] Samuel B Hopkins, Jerry Li, and Fred Zhang. Robust and heavy-tailed mean estimation made simple, via regret minimization. In *Advances in Neural Information Processing Systems 33*, NeurIPS '20, pages 11902–11912. Curran Associates, Inc., 2020.
- [Hop20] Samuel B Hopkins. Mean estimation with sub-Gaussian rates in polynomial time. *The Annals of Statistics*, 48(2):1193–1213, 2020.
- [HRRU14] Justin Hsu, Aaron Roth, Tim Roughgarden, and Jonathan Ullman. Privately solving linear programs. In *International Colloquium on Automata, Languages, and Programming*, pages 612–624. Springer, 2014.
- [HSR<sup>+</sup>08] Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V. Pearson, Dietrich A. Stephan, Stanley F. Nelson, and David W. Craig. Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PLoS Genetics*, 4(8):1–9, 2008.
- [HT10] Moritz Hardt and Kunal Talwar. On the geometry of differential privacy. In *Proceedings of the 42nd Annual ACM Symposium on the Theory of Computing*, STOC '10, pages 705–714, New York, NY, USA, 2010. ACM.
- [KLSU19] Gautam Kamath, Jerry Li, Vikrant Singhal, and Jonathan Ullman. Privately learning high-dimensional distributions. In *Proceedings of the 32nd Annual Conference on Learning Theory*, COLT '19, pages 1853–1902, 2019.
- [KLZ21] Gautam Kamath, Xingtu Liu, and Huanyu Zhang. Improved rates for differentially private stochastic convex optimization with heavy-tailed data. *arXiv preprint arXiv:2106.01336*, 2021.
- [KMS<sup>+</sup>21] Gautam Kamath, Argyris Mouzakis, Vikrant Singhal, Thomas Steinke, and Jonathan Ullman. A private and computationally-efficient estimator for unbounded gaussians. *arXiv preprint arXiv:2111.04609*, 2021.
- [KS17a] Pravesh K Kothari and Jacob Steinhardt. Better agnostic clustering via relaxed tensor norms. *arXiv preprint arXiv:1711.07465*, 2017.
- [KS17b] Pravesh K Kothari and David Steurer. Outlier-robust moment-estimation via sum-of-squares. *arXiv preprint arXiv:1711.11581*, 2017.
- [KSSU19] Gautam Kamath, Or Sheffet, Vikrant Singhal, and Jonathan Ullman. Differentially private algorithms for learning mixtures of separated Gaussians. In *Advances in Neural Information Processing Systems 32*, NeurIPS '19, pages 168–180. Curran Associates, Inc., 2019.
- [KSU20] Gautam Kamath, Vikrant Singhal, and Jonathan Ullman. Private mean estimation of heavy-tailed distributions. In *Proceedings of the 33rd Annual Conference on Learning Theory*, COLT '20, pages 2204–2235, 2020.
- [KT13] Michael Kapralov and Kunal Talwar. On differentially private low rank approximation. In *Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '13, pages 1395–1414, Philadelphia, PA, USA, 2013. SIAM.

- [KU20] Gautam Kamath and Jonathan Ullman. A primer on private statistics. *arXiv preprint arXiv:2005.00010*, 2020.
- [KV18] Vishesh Karwa and Salil Vadhan. Finite sample differentially private confidence intervals. In *Proceedings of the 9th Conference on Innovations in Theoretical Computer Science*, ITCS '18, pages 44:1–44:9, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [LKKO21] Xiyang Liu, Weihao Kong, Sham Kakade, and Sewoong Oh. Robust and differentially private mean estimation. *arXiv preprint arXiv:2102.09159*, 2021.
- [LKO21] Xiyang Liu, Weihao Kong, and Sewoong Oh. Differential privacy and robust statistics in high dimensions. *arXiv preprint arXiv:2111.06578*, 2021.
- [LM19a] Gábor Lugosi and Shahar Mendelson. Mean estimation and regression under heavy-tailed distributions: A survey. *Foundations of Computational Mathematics*, 19(5):1145–1190, 2019.
- [LM19b] Gábor Lugosi and Shahar Mendelson. Sub-Gaussian estimators of the mean of a random vector. *The Annals of Statistics*, 47(2):783–794, 2019.
- [LMV21] Jonathan Leake, Colin McSwiggen, and Nisheeth K Vishnoi. Sampling matrices from harish-chandra–itzykson–zuber densities with applications to quantum inference and differential privacy. In *Proceedings of the 53rd Annual ACM Symposium on the Theory of Computing*, STOC '21, pages 1384–1397, New York, NY, USA, 2021. ACM.
- [LRV16] Kevin A. Lai, Anup B. Rao, and Santosh Vempala. Agnostic estimation of mean and covariance. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '16, pages 665–674, Washington, DC, USA, 2016. IEEE Computer Society.
- [LSA<sup>+</sup>21] Daniel Levy, Ziteng Sun, Kareem Amin, Satyen Kale, Alex Kulesza, Mehryar Mohri, and Ananda Theertha Suresh. Learning with user-level privacy. In *Advances in Neural Information Processing Systems 34*, NeurIPS '21. Curran Associates, Inc., 2021.
- [LSY<sup>+</sup>20] Yuhan Liu, Ananda Theertha Suresh, Felix Yu, Sanjiv Kumar, and Michael Riley. Learning discrete distributions: User vs item-level privacy. In *Advances in Neural Information Processing Systems 33*, NeurIPS '20. Curran Associates, Inc., 2020.
- [MT07] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '07, pages 94–103, Washington, DC, USA, 2007. IEEE Computer Society.
- [MV21] Oren Mangoubi and Nisheeth K. Vishnoi. Sampling from log-concave distributions with infinity-distance guarantees and applications to differentially private optimization. *arXiv preprint arXiv:2111.04089*, 2021.
- [PBR19] Adarsh Prasad, Sivaraman Balakrishnan, and Pradeep Ravikumar. A unified approach to robust mean estimation. *arXiv preprint arXiv:1907.00927*, 2019.
- [PS17] Aaron Potechin and David Steurer. Exact tensor completion with sum-of-squares. In *Conference on Learning Theory*, pages 1619–1673. PMLR, 2017.

- [RSS18] Prasad Raghavendra, Tselil Schramm, and David Steurer. High dimensional estimation via sum-of-squares proofs. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*, pages 3389–3423. World Scientific, 2018.
- [SCV18] Jacob Steinhardt, Moses Charikar, and Gregory Valiant. Resilience: A criterion for learning in the presence of arbitrary outliers. In *Proceedings of the 9th Conference on Innovations in Theoretical Computer Science*, ITCS '18, pages 45:1–45:21, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [SU15] Thomas Steinke and Jonathan Ullman. Interactive fingerprinting codes and the hardness of preventing false discovery. In *Proceedings of the 28th Annual Conference on Learning Theory*, COLT '15, pages 1588–1628, 2015.
- [TVGZ20] Christos Tzamos, Emmanouil-Vasileios Vlatakis-Gkaragkounis, and Ilias Zadik. Optimal private median estimation under minimal distributional assumptions. In *Advances in Neural Information Processing Systems 33*, NeurIPS '20, pages 3301–3311. Curran Associates, Inc., 2020.
- [Wei17] Benjamin Weitz. *Polynomial proof systems, effective derivations, and their applications in the sum-of-squares hierarchy*. University of California, Berkeley, 2017.
- [WXDX20] Di Wang, Hanshen Xiao, Srinivas Devadas, and Jinhui Xu. On differentially private stochastic convex optimization with heavy-tailed data. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, pages 10081–10091. JMLR, Inc., 2020.
- [ZKKW20] Huanyu Zhang, Gautam Kamath, Janardhan Kulkarni, and Zhiwei Steven Wu. Privately learning Markov random fields. In *Proceedings of the 37th International Conference on Machine Learning*, ICML '20, pages 11129–11140. JMLR, Inc., 2020.

## A Deferred Proofs

**Lemma A.1** (SDP decreasing). SDP (Definition 6.4) is decreasing in  $r$ .

*Proof.* Suppose  $r > r'$ . Our goal is to prove that  $\text{SDP}(\tilde{\mu}, r, Z) \leq \text{SDP}(\tilde{\mu}, r', Z)$ . Suppose  $X$  is the solution that maximizes  $\text{SDP}(\tilde{\mu}, r, Z)$ . Then  $X$  is a feasible solution to  $\text{SDP}(\tilde{\mu}, r', Z)$ . Therefore  $\text{SDP}(\tilde{\mu}, r, Z) \leq \text{SDP}(\tilde{\mu}, r', Z)$ .  $\square$

**Lemma A.2.**  $\{b^2 = b\} \vdash_2 b^2 \leq 1$ .

*Proof.*

$$(1 - b)^2 + 2(b - b^2) = 1 - b^2 \implies b^2 = b \vdash_2 b^2 \leq 1.$$

$\square$

**Lemma A.3** (boundedness of the solutions of Definition 5.13). The constraints of COARSE-SDP, as defined in Definition 5.13, imply that  $\|(b, v)\|_2^2 \leq (2R + R/1000)^2 + k$ .

*Proof.* From the constraints we have that  $\|v\|_2^2 \leq (2R + R/1000)^2$ . Furthermore by Lemma A.2, we know that  $\|b\|_2^2 \leq k$ . Therefore  $\|(b, v)\|_2^2 \leq (2R + R/1000)^2 + k$ .  $\square$

**Lemma A.4** (boundedness of the solutions of Definition 6.7). The constraints of SDP-VAL, as defined in (Definition 6.3) imply that  $\|(b, v)\|_2^2 \leq k + 1$ .

*Proof.* From the constraints we have that  $\|v\|_2^2 = 1$ . Furthermore by Lemma A.2, we know that  $\|b\|_2^2 \leq k$ . Therefore  $\|(b, v)\|_2^2 \leq k + 1$   $\square$

**Lemma A.5** (QUAD-VAL( $y$ )  $\leq$  SDP-VAL( $y$ )). *For every  $Z_1, \dots, Z_k, \tilde{\mu} \in \mathbb{R}^d$ , and  $r \geq 0$ ,*

$$\text{QUAD-VAL}(y; \tilde{\mu}, r, Z) \leq \text{SDP-VAL}(y; \tilde{\mu}, r, Z).$$

*Proof.* Suppose  $(b, v)$  is a solution that maximizes the quadratic optimization problem of QUAD-VAL( $y$ ). Therefore  $\sum_{i=1}^k b_i^2 = \text{QUAD-VAL}(y)$ . Let  $X = (1, b, v) \otimes (1, b, v)$ , then  $X$  will be PSD. Suppose

$$X = \begin{bmatrix} 1 & b^\top & v^\top \\ b & B & W \\ v & W^\top & V \end{bmatrix},$$

then  $X$  has  $\text{Tr}(B) = \sum_{i=1}^k b_i^2 = \text{QUAD-VAL}(y)$ , and satisfies every constraint in the SDP-VAL optimization problem (Definition 6.7), except for  $\text{Tr}(V) = 1$ . This is because  $\|y\|_2$  may be smaller than 1. Let

$$X' = \begin{bmatrix} 1 & b^\top & v^\top \\ b & B & W \\ v & W^\top & V' = V / \text{Tr}(V) \end{bmatrix}.$$

Now the objective function's value and all of the constraints will be the same as  $X$ , except for the  $\text{Tr}(V) = 1$  constraint, which is now satisfied. It remains to argue that  $X'$  is PSD, which is a direct result of Lemma A.6.  $\square$

**Lemma A.6.** *If  $M = \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix}$  is PSD, then for any  $k \geq 1$ ,  $M' = \begin{bmatrix} A & B \\ B^\top & kC \end{bmatrix}$  is also PSD.*

*Proof.* We prove the third condition in Lemma B.6 for  $M'$ . Note that  $kC^\dagger = C^\dagger/k$ . From  $M \succcurlyeq 0$ ,

- $A \succcurlyeq 0$ ,
- $C \succcurlyeq 0$ ,
- $(I - CC^\dagger)B^\top = 0$ ,
- $A - BC^\dagger B^\top \succcurlyeq 0$ ,

and we want to prove that

- $kC \succcurlyeq 0$ ,
- $(I - kC(kC)^\dagger)B^\top = 0$ ,
- $A - B(kC)^\dagger B^\top \succcurlyeq 0$ .

The first two are obvious. For the last one, note that

$$A \succcurlyeq 0 \implies (k-1)A \succcurlyeq 0 \implies kA \succcurlyeq A.$$

Therefore

$$kA \succcurlyeq A \succcurlyeq BC^\dagger B^\top,$$

and

$$A - BC^\dagger B^\top / k \succcurlyeq 0,$$

as desired.  $\square$



## B Lemmata

**Lemma B.1.** *Suppose  $M \succcurlyeq 0$  is a matrix in  $\mathbb{R}^{d \times d}$ ,  $1 \leq k \leq d$ . Let  $N$  be a  $d \times d$  matrix such that for every  $1 \leq i, j \leq d$ ,  $N_{i,j} = M_{i,j}$ , if  $i \neq k$  and  $j \neq k$ , and  $N_{i,j} = 0$ , if  $i = k$ , or  $j = k$ . Then  $N \succcurlyeq 0$ .*

*Proof.*

$$\begin{aligned} \min_{\|z\|_2=1} z^\top N z &= \min_{\|z\|_2=1, z_k=0} z N^\top z \\ &= \min_{\|z\|_2=1, z_k=0} z M^\top z \\ &\geq \min_{\|z\|_2=1} z M^\top z \\ &\geq 0 \end{aligned}$$

□

**Lemma B.2.** *Suppose  $\mathcal{K}$  is a closed convex set, then  $f(y) = \max_{\substack{x \in \mathcal{K} \\ Ax=y}} \langle x, c \rangle$  is concave.*

*Proof.* We need to prove that for  $\alpha \in [0, 1]$

$$\alpha f(y_1) + (1 - \alpha) f(y_2) \leq f(\alpha y_1 + (1 - \alpha) y_2).$$

Suppose  $f(y_1)$  is maximized at  $x_1$  and  $f(y_2)$  is maximized at  $x_2$ . Let  $z = \alpha x_1 + (1 - \alpha) x_2 \in \mathcal{K}$ , then  $Az = \alpha y_1 + (1 - \alpha) y_2$  and

$$f(\alpha y_1 + (1 - \alpha) y_2) \geq \langle z, c \rangle,$$

which implies the desired inequality. □

The following lemma is the ordinary binary search.

**Lemma B.3** (Non Private Binary Interval Search). *Suppose an interval  $[0, D]$  is given. For an interval  $[s, e] \subseteq [0, D]$ , and for an accuracy parameter  $a$ , the binary interval search algorithm returns a point  $x_T$  after  $T = \log(D/a)$  steps of the following binary search, where  $x_i$  is the parameter of the binary search at step  $i$ .*

1. If  $x_i < s$ , move right.
2. If  $s \leq x_i \leq e$ , move in an arbitrary direction. Note that naturally in a binary search we would terminate in this step, but here we assume that we might go to any direction.
3. if  $x_i > e$ , move left.

Then  $x_T \in [s - a, e + a]$ .

**Lemma B.4** (Cauchy-Schwarz inequality). *If  $(f, g)$  is a level-2 fictitious random variable over  $\mathbb{R}^u \times \mathbb{R}^u$ , then*

$$\tilde{\mathbf{E}}_{f,g} \langle f, g \rangle \leq \sqrt{\tilde{\mathbf{E}}_f \|f\|_2^2} \cdot \sqrt{\tilde{\mathbf{E}}_g \|g\|_2^2}.$$

**Lemma B.5** (pseudo-covariance). *If  $\mathcal{X}$  is a degree-2 pseudo-distribution over  $\mathbb{R}^d$ , then*

$$\tilde{\mathbf{E}}_{\mathcal{X}} \left[ \left\| x - \tilde{\mathbf{E}}_{\mathcal{X}} [x] \right\|_2^2 \right] = \tilde{\mathbf{E}}_{\mathcal{X}} [\|x\|_2^2] - \|\tilde{\mathbf{E}}_{\mathcal{X}} [x]\|_2^2.$$

*Proof.*

$$\begin{aligned}
\tilde{\mathbf{E}}_{\mathcal{X}} \left[ \left\| x - \tilde{\mathbf{E}}_{\mathcal{X}} [x] \right\|_2^2 \right] &= \sum_i \tilde{\mathbf{E}}_{\mathcal{X}} \left[ \left( x_i - \tilde{\mathbf{E}}_{\mathcal{X}} [x_i] \right)^2 \right] \\
&= \sum_i \tilde{\mathbf{E}}_{\mathcal{X}} [x_i^2] - \left( \tilde{\mathbf{E}}_{\mathcal{X}} [x_i] \right)^2 \\
&= \tilde{\mathbf{E}}_{\mathcal{X}} [\|x\|_2^2] - \|\tilde{\mathbf{E}}_{\mathcal{X}} [x]\|_2^2
\end{aligned}$$

□

**Lemma B.6** (PSD  $2 \times 2$  block matrix). *Given any matrix,  $M = \begin{pmatrix} A & B \\ B^\top & C \end{pmatrix}$ , the following conditions are equivalent:*

1.  $M \succcurlyeq 0$  ( $M$  is positive semidefinite).
2.  $A \succcurlyeq 0$ ,  $(I - AA^\dagger)B = 0$ ,  $C - B^\top A^\dagger B \succcurlyeq 0$ .
3.  $C \succcurlyeq 0$ ,  $(I - CC^\dagger)B^\top = 0$ ,  $A - BC^\dagger B^\top \succcurlyeq 0$ .

**Fact B.7** (SoS Triangle Inequality). *Let  $x, y$  be indeterminates. Let  $t$  be a power of 2. Then*

$$\vdash_t (x + y)^2 \leq 2^{t-1}(x^t + y^t).$$

**Fact B.8** (lower bound for spherical caps [Bal97]). *For  $0 \leq r \leq 2$ , a cap of radius  $r$  on  $\mathbb{S}^{n-1}$  has measure at least  $\frac{1}{2}(r/2)^{n-1}$ .*