2021 IEEE Symposium on Security and Privacy (SP)

Manipulation Attacks in Local Differential Privacy

Albert Cheu
Khoury College of Computer Sciences
Northeastern University
Boston, Massachusetts
cheu.a@northeastern.edu

Adam Smith
Department of Computer Science
Boston University
Boston, Massachusetts
ads22@bu.edu

Jonathan Ullman
Khoury College of Computer Sciences
Northeastern University
Boston, Massachusetts
jullman@ccs.neu.edu

2021 IEEE Symposium on Security and Privacy (SP) | 978-1-7281-8934-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/SP40001.2021.00001

***Abstract*—Local differential privacy is a widely studied restriction on distributed algorithms that collect aggregates about sensitive user data, and is now deployed in several large systems. We initiate a systematic study of a fundamental limitation of locally differentially private protocols: *they are highly vulnerable to adversarial manipulation.* While any algorithm can be manipulated by adversaries who lie about their inputs, we show that any noninteractive locally differentially private protocol can be manipulated to a much greater extent—when the privacy level is high, or the domain size is large, a small fraction of users in the protocol can completely obscure the distribution of the honest users' input. We also construct protocols that are optimally robust to manipulation for a variety of common tasks in local differential privacy. Finally, we give simple experiments validating our theoretical results, and demonstrating that protocols that are optimal without manipulation can have dramatically different levels of robustness to manipulation. Our results suggest caution when deploying local differential privacy and reinforce the importance of efficient cryptographic techniques for the distributed emulation of centrally differentially private mechanisms.**

I. Introduction

Many companies rely on aggregates and models computed on sensitive user data. The past few years have seen a wave of deployments of systems for collecting sensitive user data via *local differential privacy* [1], notably Google's RAPPOR [2] and Apple's deployment in *iOS* [3]. These protocols satisfy differential privacy [4], a widely studied restriction that limits the information leaked due to any one user's presence in the data. Furthermore, the privacy guarantee is enforced *locally*, by a user's device, without reliance on the correctness of other parts of the system. See Figure 1 for a diagram.

Local differential privacy is attractive for deployments for several reasons. The trust assumptions are relatively weak and easily explainable to novice users. In contrast to centralized differential privacy, the data collector never collects raw data, reducing the legal, ethical, and technical burden of safeguarding the data. Moreover, local protocols are typically simple and highly efficient in terms of communication and computation.

Despite these benefits, local protocols have significant limitations when compared to private algorithms in the *central model*, in which data are collected and processed by a trusted curator. The most discussed limitation is larger error for the same level of privacy (e.g., [4, 5, 6]). In this paper, we initiate a systematic study of a different limitation that we show to be equally fundamental:

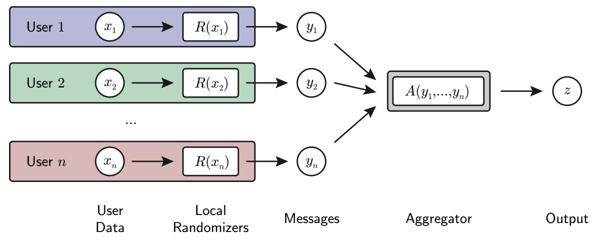

Fig. 1: The structure of a (noninteractive) local protocol.

> *Locally differentially private protocols*
> *are highly vulnerable to manipulation.*

While any algorithm can be manipulated by users who lie about their data, we demonstrate that local algorithms can be manipulated to a far greater extent. As the level of privacy or the size of the input domain increase, an adversary who corrupts a vanishing fraction of the users can effectively prevent the protocol from collecting any useful information about the data of the honest users. This result can be interpreted as showing that local differential privacy opens up new, more powerful avenues for *poisoning attacks*—poisoning the private messages can be far more destructive than poisoning the data itself.

Various attackers might be able to exploit this vulnerability to manipulation for nefarious purposes. In particular, if a company is using locally differentially private protocols to collect user data that it then uses to improve its product, then its rivals would have an incentive to exploit these vulnerabilities to gain a competitive edge. If the goal is distribution estimation, our work implies that the rival only needs to corrupt a small fraction of users to highly skew the estimate in statistical distance. Furthermore, we find a setting where estimates can be vulnerable to a *small number* of corruptions.

Prior work had already noted that a *specific* protocol—Warner's randomized response [7]—is vulnerable to manipulation [8, 9]. A concurrent and independent work [10] gives

AC and JU were supported by NSF grants CCF-1718088, CCF-1750640, and CNS-1816028. JU was also supported by a Google Faculty Research Award. AS was supported by NSF award CCF-1763786 and a Sloan Foundation Research Award. The full version of this paper is accessible at https://arxiv.org/abs/1909.09630.

DOI 10.1109/SP40001.2021.00001

an empirical study the effectiveness of natural manipulation attacks against common protocols. In contrast, we show that manipulation is unavoidable for *any* noninteractive local protocol that solves any one of a few basic problems to sufficiently high accuracy, and systematically identify the optimal degree of manipulation for each problem. These problems capture computing means and histograms, identifying heavy-hitters, and estimating the distribution of users' data. In particular, our work is the first to identify the domain size as a key factor in determining how vulnerable local protocols must be to manipulation. We also give simple experiments validating our theoretical findings. In addition, these experiments show that two protocols that have exactly identical error absent manipulation can nonetheless have dramatically different performance in the presence of manipulation.

Our results suggest caution when deploying locally differentially private protocols: the architecture is *inherently* vulnerable to manipulation. One way to remedy this is to introduce some mechanism that enforces the correctness of users' randomization, such as physical constraints or an interactivity requirement [9, 8]. Our work also reinforces the importance of efficient cryptographic techniques that emulate central-model algorithms in a distributed setting, such as multiparty computation [11] or shuffling [12, 13]. Such protocols already have significant accuracy benefits, and our results highlight their much greater resilience to manipulation.

A. *Why are Local Protocols Vulnerable to Manipulation?*

Intuitively, because local differential privacy requires that each user's message is almost independent of their data, large changes in the users' data induce only small changes in the distribution of the messages. As a result, the aggregator must be highly sensitive to small changes in the distribution of messages. That is, an adversary who can cause small changes in the distribution of messages can make the messages appear as if they came from users with very different data, forcing the aggregator to change its output dramatically.

We can see how this occurs using the classic *randomized response* protocol. Here, each user's has data $x_i \in \{\pm 1\}$ and the objective is to estimate the mean $\frac{1}{n}\sum_{i=1}^{n} x_i$. For roughly 2ε-local differential privacy, each user outputs

$$y_i = \begin{cases} x_i & \text{with probability } \frac{1+\varepsilon}{2} \\ -x_i & \text{with probability } \frac{1-\varepsilon}{2} \end{cases}$$

so that $\mathbb{E}[y_i] = \varepsilon x_i$. The aggregator computes an unbiased estimate of the mean by returning $\frac{1}{n}\sum_{i=1}^{n} \frac{y_i}{\varepsilon}$.

In order to extract the relatively weak signal and make the estimate unbiased, the aggregator scales up each message y_i by a factor of $\frac{1}{\varepsilon}$, which increases the influence of each message. This means that an adversary who can flip m of the messages y_i from -1 to $+1$ will increase the aggregator's output by $\frac{2m}{\varepsilon n}$. A simple consequence of our work is that *any* noninteractive LDP protocol for computing the average of bits is similarly vulnerable to manipulation.

B. *Frequency Estimation: A Representative Example*

We can more fully illustrate our work results through the example of *frequency estimation*. Consider a protocol whose goal is to collect the frequency of words typed by users on their keyboard. We assume that there are n users, and each user contributes only a single word to the dataset, so each user's word is an element of $[d] = \{1, \ldots, d\}$ where d is the size of the dictionary. The goal of the protocol to estimate the vector consisting of the frequency of each word as accurately as possible. In this example, we measure accuracy in the ℓ_1 norm (or, equivalently, in statistical distance or total variation distance): if $v \in \mathbb{R}^d$ is the frequency vector whose entries v_j are the fraction of users whose data takes the value j, and $\hat{v}$ is the estimated frequency vector, then the error is $\|v - \hat{v}\|_1 = \sum_{j=1}^{d} |v_j - \hat{v}_j|$.

Baseline Attacks. In order for the attack to be a concern, the adversary has to be able to introduce more error than what would otherwise exist in the protocol, and the attack should be specific to local differential privacy. In particular, we say the attack is nontrivial if it introduces more error than the following trivial *baselines*:

No Manipulation. The adversary could choose not to manipulate the messages at all, in which case the protocol will still incur some error due to the fact that it must ensure local differential privacy. For example, it is known that an optimal ε-differentially private local protocol for frequency estimation introduces error $\approx \sqrt{d^2/\varepsilon^2 n}$ [14].

Input Manipulation. The adversary could have the corrupted users change only their inputs. That is, the corrupted users could honestly carry out the protocol as if their data were some arbitrary x_i' instead of x_i (see Figure 2). Since the corrupted users control an m/n fraction of the data, they can skew the overall distribution by m/n. This attack applies to *any* protocol, private or not.

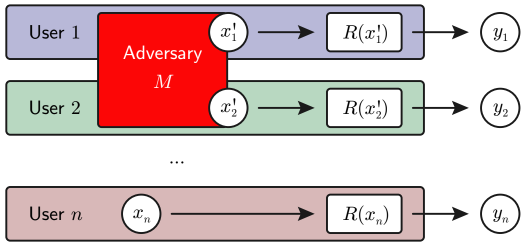

Fig. 2: An input-manipulation attack.

These baselines make sense in the context of any task, and we will use the bounds for these baselines to calibrate the effectiveness of attacks for other problems (not just frequency estimation) in the next section.

Our Work: Manipulation Attacks. We consider a general attack model where the adversary is able to corrupt a set of m

out of the n users' devices, and can instruct these users to send arbitrary messages, possibly in a coordinated fashion; we visualize this model in Figure 3. The corruptions are unknown to the aggregator running the protocol to prevent the aggregator from ignoring the messages of the corrupted users. In this, and all of our examples, the adversary's goal is to make the error as large as possible—exactly opposite to goal of the protocol.

In Section IV, we describe and analyze an attack that skews the overall distribution by $\approx \frac{m\sqrt{d}}{\varepsilon n}$, for *any* noninteractive ε-differentially private local protocol. This attack introduces much larger error—by about a $\frac{\sqrt{d}}{\varepsilon}$ factor—than input manipulation, and thus shows specifically that locally private protocols are highly vulnerable to manipulation. We also show our attack is near-optimal by giving a protocol that achieves optimal error in the absence of manipulation and cannot be manipulated by more than $\approx \frac{m\sqrt{d}}{\varepsilon n}$.

For comparison, an adversary of a centrally private algorithm is limited to input manipulation. This is because each user communicates their data noiselessly: in the mean estimation example, the aggregator has no need to increase the influence of each user. Additionally, techniques that simulate centrally private algorithms in a distributed setting such as multiparty computation and shuffling can inherit this resilience.

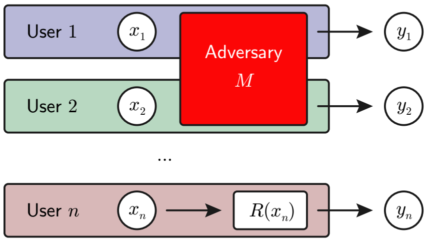

Fig. 3: A general manipulation attack.

Measuring the Effectiveness of Attacks. In this work we establish tight upper and lower bounds on the error introduced by manipulation in terms of the parameters n, m, ε, and d. To reduce the number of parameters, and facilitate easier comparisons to the baseline attacks, we have identified two key thresholds that we can use to understand the effectiveness of manipulation attacks for a given task.

The first is what we call the *breakdown point*, which is the minimum fraction of users at which the protocol can no longer guarantee non-trivial accuracy. For all problems we consider, the accuracy is non-trivial if it is smaller than some fixed constant (where the choice of constant will not affect the asymptotic bounds). Our attack demonstrates that, for frequency estimation, the breakdown point is roughly $\frac{\varepsilon}{\sqrt{d}}$. That is, that this number of corrupted users can skew the distribution by $\Omega(1)$ in ℓ_1 norm, while *any* two frequency vectors have ℓ_1 distance at most 2. Thus, when ε is small or d is large, an attacker controlling a vanishing fraction of the users can prevent the protocol from achieving any nontrivial accuracy guarantee.

The second threshold is what we call the *significance point*, which is the minimum fraction of users that can increase the error significantly beyond the error necessary to solve the problem absent manipulation. That is, the corrupted users can introduce error on the same order as the error of an optimal protocol with no manipulation. For the frequency estimation problem, the optimal error absent manipulation is $\sqrt{d^2/\varepsilon^2 n}$, and thus the significance point is $\sqrt{d/n}$.

C. Summary of Results: Lower Bounds

In this work, we construct two manipulation attacks on locally differentially private protocols, and use these attacks to derive lower bounds on the degree of manipulation allowed by local protocols for a variety of tasks (including the frequency estimation example above). We also study the resilience of specific protocols to manipulation. For each problem, we give a protocol that is asymptotically optimal with respect to both ordinary accuracy (i.e., without manipulation) and resilience to manipulation. We also show that popular protocols for most tasks are much less resistant to manipulation than optimal ones.

Below, we first discuss the attacks informally, and then discuss the set of problems to which they apply. We defer details of the attack model to Section II-B. Our results are summarized in Table I.

An Attack for Binary Data. Our first attack concerns the simplest problem in local differential privacy—computing a mean of bits. Each user has data $x_i \in \{0, 1\}$, and we assume that each x_i is drawn independently from the Bernoulli distribution $\mathbf{Ber}(p)$, meaning $x_i = 1$ with probability p and $x_i = 0$ with probability $1 - p$. Our goal is to estimate the mean p as accurately as possible. More generally, we could allow the users to have arbitrary data $x_1, \ldots, x_n \in \{0, 1\}$ and try to estimate $\frac{1}{n}\sum_{i=1}^{n} x_i$. For the purposes of attacks, considering the distributional version only makes our results stronger.

Without manipulation, this problem is solved by the classical randomized response protocol [7], which achieves optimal error $\Theta(\frac{1}{\varepsilon\sqrt{n}})$. As we discussed in the introduction, one can show that the error of randomized response increases to $\Theta(\frac{1}{\varepsilon\sqrt{n}} + \frac{m}{\varepsilon n})$ when an adversary corrupts m of the users. We show that no protocol can improve this bound.

Theorem I.1 (Informal)**.** *For every ε-differentially private local protocol Π for n users with input domain $\{0, 1\}$, there is an attack M corrupting m users such that Π cannot distinguish between the following cases:*

1) *The data is drawn from $\mathbf{Ber}(p_0)$ for $p_0 = \frac{1}{2}$ and Π has been manipulated by M.*
2) *The data is drawn from $\mathbf{Ber}(p_1)$, p_1 and Π has not been manipulated where*

$$p_1 - p_0 = \frac{e^\varepsilon + 1}{e^\varepsilon - 1} \cdot \frac{m}{2n} = \Theta\big(\frac{m}{\varepsilon n}\big)$$

This theorem—combined with existing lower bounds for locally differentially private estimation—shows that, when the data is drawn from $\mathbf{Ber}(p)$ for unknown p, no protocol Π can estimate p and guarantee accuracy better than $\Theta(\frac{1}{\varepsilon\sqrt{n}} + \frac{m}{\varepsilon n})$.

As an immediate consequence, when the data $x_1, \ldots, x_n \in \{0,1\}$ may be arbitrary, no protocol Π can estimate the mean $\frac{1}{n}\sum_i x_i$ with significantly better accuracy. Concretely, the $\frac{e^\varepsilon+1}{e^\varepsilon-1} \cdot \frac{m}{2n}$ error due to manipulation is within a factor of 4 of the upper bound that can be proved for randomized response. We have not attempted to optimize this constant factor, and we would conjecture that randomized response has exactly optimal robustness to manipulation.

Attacks for Large Domains. Since estimating the mean of bits is a special case of most problem studied in the local model, this attack already shows that manipulation can cause additional error of $\Omega(\frac{m}{\varepsilon n})$ for many problems. In some cases, this bound is already near-optimal, and some protocol achieves a similar upper bound. However, for many cases of interest (such as the frequency estimation example), protocols become more vulnerable to manipulation when the size of the input domain increases. Our second result is an attack on any protocol accepting inputs from the domain $[d] = \{1, \ldots, d\}$ for large d, showing that manipulation can skew the distribution by $\tilde{\Omega}(\frac{m\sqrt{d}}{\varepsilon n})$ without being detected.

Theorem I.2 (Informal)**.** *For every ε-differentially private local protocol Π for n users with input domain $[d]$, there is an attack M corrupting m users such that Π cannot distinguish between the following cases:*

1) *The data is drawn from the uniform distribution $\mathbf{U}$ over $[d]$ and M manipulates Π.*
2) *The data is drawn from some distribution $\mathbf{P}$ over $[d]$ with $\|\mathbf{U} - \mathbf{P}\|_1 = \Theta(\frac{1}{\varepsilon}\sqrt{\frac{d}{\log n}}(\frac{1}{\sqrt{n}} + \frac{m}{n}))$ and Π has not been manipulated.*

For a large class of natural protocols, the bound on $\|\mathbf{U} - \mathbf{P}\|_1$ can be sharpened to $\Theta(\frac{\sqrt{d}}{\varepsilon}(\frac{1}{\sqrt{n}} + \frac{m}{n}))$.

A consequence of this attack for the example of frequency estimation above is that any local protocol can have the distribution skewed by $\tilde{\Omega}(\frac{m\sqrt{d}}{\varepsilon n})$. As we show in Section V, this bound is actually matched by a simple protocol. In order to simplify the proof and obtain a statement that applies to arbitrary protocols, we do not optimize the constant factors hidden by the $\Theta(\cdot)$ notation. However, we do give proof-of-concept experiments in Section VI showing the concrete effect of our attack on a widely studied frequency estimation protocol.

D. Summary of Results: Optimal Protocols

We consider a variety of tasks of interest in local differential privacy. For each, we show that one of the attacks above gives an optimal bound on the vulnerability of protocols for that task. The results are summarized in Table I.

Most tasks we consider can be formulated as instances of the following *ℓ_p/ℓ_q-mean estimation* problem for vectors in $\mathbb{R}^d$. Each user's data x_i is a vector in $\mathbb{R}^d$ such that the ℓ_p-norm of each data point is bounded, $\|x_i\|_p \leq 1$. The protocol's goal is to output an estimate of the mean $\hat{\mu}$ with low error in the ℓ_q-norm, $\|\hat{\mu} - \frac{1}{n}\sum_{i=1}^n x_i\|_q$. Recall that $\|v\|_p = (\sum_i v_i^p)^{1/p}$ and $\|v\|_\infty = \max_i |v_i|$. This setup captures a number of widely studied problems:

- The frequency estimation example above is a special case of ℓ_1/ℓ_1 estimation, where each user represents their word $x_i \in [d]$ by the standard basis vector $e_{x_i} \in \mathbb{R}^d$ with a 1 in the x_i-th coordinate and 0 elsewhere.
- Computing a histogram of data in $[d]$ is a special case of ℓ_1/ℓ_∞-mean estimation. The *heavy-hitters (HH)* problem, which asks one only to identify the heaviest bins of a histogram and their frequencies, suffices to solve ℓ_1/ℓ_∞-mean estimation, so manipulation attacks on the latter thus imply attacks on the former. Computing heavy-hitters has been a focus of the past few years [18, 16, 19, 20], and it is central to systems deployed by Google and Apple [2, 3].
- Computing the answers to d statistical queries [21, 22, 5] is a special case of ℓ_∞/ℓ_∞-mean estimation. Users have data in some arbitrary domain $\mathcal{X}$, there are d query functions $f_1, \ldots, f_d : \mathcal{X} \to [-1,1]$, and we would like an accurate estimate of $\sum_{i=1}^n f_j(x_i)$ for every j. In the corresponding mean estimation instance, $x_i = (f_1(x_i), \ldots, f_d(x_i))$.
- When minimizing a sum of convex functions $f(\theta) = \sum_{i=1}^n f_{x_i}(\theta)$ defined by the users' data (e.g. to train a machine learning model), one often computes the average gradient $\sum_{i=1}^n \nabla f_{x_i}(\theta_t)$ at a sequence of points θ_t. Typically one assumes that the gradients are bounded in ℓ_2, and convergence requires an accurate estimate in ℓ_2, making this an instance of ℓ_2/ℓ_2-mean estimation. (More generally, optimization *requires* this sort of estimation [23]).
- We study one further problem, *ℓ_1/ℓ_1-uniformity testing*, for which Acharya et al. [15] gave optimal LDP protocols. Assuming the data is drawn from some distribution over $[d]$, we want to determine if this distribution is either uniform or is far from uniform in ℓ_1 distance.

Since every ℓ_p/ℓ_q mean estimation problem generalizes binary mean estimation (the special case where $d = 1$), our first attack gives a lower bound on all of these problems. Our second attack is precisely an attack on the ℓ_1/ℓ_1-testing problem, and thus implies a lower bound of $\tilde{\Omega}(\frac{m\sqrt{d}}{\varepsilon n})$ for that problem. Finally, since ℓ_1/ℓ_1-mean estimation problem strictly generalizes ℓ_1/ℓ_1-testing problem—once we estimate the mean, we can determine if it is close to uniform or far from uniform—we obtain the same lower bound for that problem.

For all of these problems we also identify and analyze protocols whose error nearly matches the lower bounds established by our attacks. These protocols generally use the public-coin model to compress each player's report to a single bit, thus reducing their influence.

E. Overview of Techniques

Attack for Binary Data. Our argument boils down to proving the following claim: for every ε-differentially private local protocol, there is some attacker who corrupts each user independently with probability $\frac{m}{n}$ in such a way that data drawn from $\mathbf{Rad}(0)$ appears as if it were drawn from $\approx \mathbf{Rad}(\frac{m}{\varepsilon n})$. To show this, we rely on a lemma from Kairouz et al. [24], which implies that for any ε-differentially private local randomizer R, the distribution

Problem	No Manipulation	Manipulation UB	Manipulation LB	Breakdown Point	Significance Point
ℓ_1/ℓ_1 Estimation (Frequency Estimation)	$\Theta(\sqrt{\frac{d^2}{\varepsilon^2 n}})$ [14]	$\tilde{O}(\frac{m}{n} \cdot \frac{\sqrt{d}}{\varepsilon})$ Thm V.7	$\Omega(\frac{m}{n} \cdot \frac{\sqrt{d}}{\varepsilon\sqrt{\log n}})$ ♭ Thm IV.9	$O\left(\varepsilon\sqrt{\frac{\log n}{d}}\right)$ ♭	$O\left(\sqrt{\frac{d \log n}{n}}\right)$ ♭
ℓ_1/ℓ_1 Testing (Uniformity Testing)	$\Theta(\sqrt{\frac{d}{\varepsilon^2 n}})$ [15]	$O(\frac{m}{n} \cdot \frac{\sqrt{d}}{\varepsilon})$ Thm V.9	$\Omega(\frac{m}{n} \cdot \frac{\sqrt{d}}{\varepsilon\sqrt{\log n}})$ ♭ Thm IV.8	$O\left(\varepsilon\sqrt{\frac{\log n}{d}}\right)$ ♭	$O\left(\sqrt{\frac{\log n}{n}}\right)$ ♭
ℓ_1/ℓ_∞ Estimation (Histograms / HH)	$\Theta(\sqrt{\frac{\log d}{\varepsilon^2 n}})$ [16]	$O(\frac{m}{n} \cdot \frac{\log d}{\varepsilon})$ Thm B.1	$\Omega(\frac{m}{n} \cdot \frac{1}{\varepsilon})$ Thm III.4	$O(\varepsilon)$	$O\left(\sqrt{\frac{\log d}{n}}\right)$
ℓ_∞/ℓ_∞ Estimation (d Statistical Queries)	$\Theta(\sqrt{\frac{d \log d}{\varepsilon^2 n}})$ [Folklore]	$O(\frac{m}{n} \cdot \frac{1}{\varepsilon})$ ♯ Thm V.2	$\Omega(\frac{m}{n} \cdot \frac{1}{\varepsilon})$ Thm III.4	$O(\varepsilon)$	$O\left(\sqrt{\frac{d \log d}{n}}\right)$
ℓ_2/ℓ_2 Estimation (Gradients)	$\Theta(\sqrt{\frac{d}{\varepsilon^2 n}})$ [17]	$\tilde{O}(\frac{m}{n} \cdot \frac{1}{\epsilon})$ Thm V.8	$\Omega(\frac{m}{n} \cdot \frac{1}{\varepsilon})$ Thm III.4	$O(\varepsilon)$	$O\left(\sqrt{\frac{d}{n}}\right)$

TABLE I: Summary of Results. In each case, [No Manipulation] is the optimal error achievable under local differential privacy without manipulation. For each problem, we identify some protocol that has optimal error without manipulation such that manipulation can increase the error by [Manipulation UB] and show that manipulation can make the error of any local protocol as large as [Manipulation LB]. In each case, no protocol can guarantee nontrivial accuracy in the presence of [Breakdown Point] corrupted users, and the error can be asymptotically increased in the presence of [Significance Point] corrupted users. ♯ indicates that the upper bound limited to public-string-oblivious attacks. ♭ indicates that the $\sqrt{\log n}$ factor can be removed for a natural class of protocols. In all cases, input-manipulation influences the output by $\frac{m}{n}$, and we present the upper and lower bounds as multiples of that baseline.

$R(\mathbf{Rad}(\mu))$ is exactly a mixture $R^{(\mu)}$ of two distributions R^+ and R^-, and

$$R^{(\mu)} \approx \tfrac{1+\varepsilon\mu}{2} \cdot R^+ + \tfrac{1-\varepsilon\mu}{2} \cdot R^- .$$

Since the data and messages are independent and identically distributed (iid), the messages consist of n iid samples from $R^{(\mu)}$. If $\mu = 0$, but an attacker corrupts each user independently with probability $\frac{m}{n}$, and has the corrupted users send a message sampled from R^+, then the messages remain independent and consist of n messages sampled from $R^{(\mu)}$ for $\mu = \frac{m}{\varepsilon n}$, exactly the same if there were no corruptions but $\mu = \frac{m}{\varepsilon n}$. Since the aggregator cannot distinguish these two identical distributions, it must have error at least $\approx \frac{m}{2\varepsilon n}$ on one of them. Some technicalities arise in the proof because (1) the attacker has a fixed budget of m corruptions that might be exceeded when corrupting each user independently, and (2) the local randomizer might only satisfy (ε, δ)-differential privacy, and thus might have a slightly more complex structure.

Attack for High-Dimensional Data. For high-dimensional data, we can show the existence of a distribution R^+ that has an even more extreme effect on the overall distribution of messages than in the binary case. For any distribution S on the domain $\{1, \ldots, d\}$, let $R^{(S)}$ be the distribution on messages $R(S)$. Let U be the uniform distribution on the domain. We show (roughly) that for every ε-differentially private local rnadomizer R, there is some distribution S supported on $d/2$ domain elements, such that $R^{(U)}$ and $R^{(S)}$ are only $\varepsilon/\sqrt{d}$ apart, and there exists an extreme distribution $R^+ \approx R^{(U)} + \frac{1}{\varepsilon}(R^{(S)} - R^{(U)})$. Thus, if we corrupt only about an $\varepsilon/\sqrt{d}$ fraction of users, we can make messages from $R^{(U)}$ look like messages from $R^{(S)}$. Since S and U have distnace at least $1/2$, corrupting about an $\varepsilon/\sqrt{d}$ fraction of users is enough to make the error at least $1/2$. With some rescaling we can prove the bound that we claim for an arbitrary number of corruptions. For technical reasons, our formal proof works by a reduction to the binary attack, in which we argue that any ε-differentially private protocol for frequency estimation can be used to get a protocol for binary estimation that is approx $(\varepsilon/\sqrt{d})$-differentially private.

Optimally Robust Protocols. All of the optimally robust protocols we present have already appeared in the literature, but had not been analyzed with respect to manipulation attacks. However, not all protocols with optimal accuracy without manipulation have optimal robustness to manipulation. In particular, the protocols that we show are optimally robust use the public-coin model to reduce the amount of communication per user down to a single bit (see e.g. [5, 16]), and thereby dramatically decreases the space of possible manipulation.

F. Related Work

Manipulation Attacks. Prior work had already observed that the specific randomized response protocol was vulnerable to manipulation [8, 9]. In contrast to ours, these works constructed efficient cryptographic protocols for sampling from the correct distribution, which resist our attacks. Our work shows that some degree of cryptography is necessary to avoid manipulation. A concurrent and independent work [10] performed an empirical study of simple manipulation attacks on common protocols for tasks like frequency estimation and heavy-hitters. In contrast to ours, their work does not prove any inherent limitations on the robustness of local protocols to manipulation, nor does it establish the crucial role that the domain size plays.

Our work is loosely related to *data poisoning attacks* in adversarial machine learning. In data poisoning, the adversary is inserts additional data to somehow degrade the quality of the output. Our attacks can be viewed as data poisoning attacks where the "data" being poisoned is actually the messages to the protocol. Thus, our results can be viewed as showing that adding local randomization to achieve privacy makes the protocol much more vulnerable to data poisoning.

Cryptographic Approaches. Our work reinforces the importance of efficient cryptographic techniques that emulate central-model algorithms in a distributed setting. Multiparty computation (MPC) allows a network of parties to jointly execute a randomized algorithm on encrypted or secret-shared data while exposing only the final result of the computation. The value of simulating a differentially private computation was first highlighted in [11, 6]. Briefly, the MPC approach gets the accuracy of the central model, and limits attackers to input manipulation, which is unavoidable without some outside certification of inputs. The downside of this approach is computational efficiency. Despite recent advances in practical MPC, applications like collecting information about mobile data usage place extreme demands on protocols that make current solutions difficult to use. To our knowledge, known MPC protocols either scale poorly to large networks, assume an honest-but-curious server (e.g., [25]), or leak extra, hard-to-reason-about intermediate results from a computation. Although the MPC literature is too vast to survey here, we refer the reader to a recent survey of the issues that arise in *federated learning* for a [26] more thorough discussion of these issues.

One recent approach asks whether we can reduce important differentialy private algorithms to some simple primitive which is easier to implement in MPC. For example, the *shuffled model* [12, 13, 27] assumes the availability of a trusted shuffling primitive, which anonymizes the origin of the messages by applying a secret permutation before delivering them to the aggregator. That model allows accuracy close to that of the central model for several tasks but leaves open just how well the shuffler can be implemented by a real protocol. On the other hand, shuffled protocols for histograms are more resilient than counterparts in the local model. [13], for example, give a protocol where the influence of each message is scaled by a factor close to 1 instead of $\frac{1}{\varepsilon}$ as in the local model.

Finally, cryptographic protocols can be used in a much narrower and potentially scalable way to ensure that local-model protocols are carried out without manipulation (see [8] for a protocol tailored to binary randomized response). These require some interaction between clients and the server and retain the accuracy limitations of the local model, but can constrain the client to simple input manipulation. Specific physical devices, such as carefully generated scratch cards, can also provide such a guarantee [9]. Current techniques for efficient MPC should suffice for wider use of such protocols.

G. Organization

In Section II we introduce the model and key concepts. In Section III, we demonstrate attacks on protocols for binary data, and in Section IV, we demonstrate attacks on protocols for large data domains. In Section V we identify protocols with near-optimal resistance to manipulation for a variety of canonical problems in local differential privacy. In Section VI we present our experiments with our attack on natural protocols for frequency estimation.

II. Threat Model and Preliminaries

A. Local Differential Privacy

In this model there are n *users*, and each user $i \in [n]$ holds some sensitive data $x_i \in \mathcal{X}$ belonging to some *data universe* $\mathcal{X}$. There is also a public random string S. Finally there is a single *aggregator* who would like to compute some function of the users' data $x_1, \dots, x_n$. In this work, for simplicity, we restrict attention to *non-interactive local differential privacy*, meaning the users and the aggregator engage in the following type of protocol:

1) A public random string S is chosen from some distribution $\mathbf{S}$ over support $\mathcal{S}$.
2) Each user computes a *message* $y_i \leftarrow R_i(x_i, b)$ using a *local randomizer* $R_i : \mathcal{X} \times \mathcal{S} \to \mathcal{Y}$.
3) The aggregator $A : \mathcal{Y}^n \times \mathcal{S} \to \mathcal{Z}$ computes some output $z \leftarrow A(y_1, \dots, y_n, S)$.

Thus the protocol Π consists of the tuple $\Pi = ((R_1, \dots, R_n), A, \mathbf{S})$. We will sometimes write $\vec{R}$ to denote the local randomizers $(R_1, \dots, R_n)$. If $R_1 = \cdots = R_n = R$ then we say the protocol is *symmetric* and denote it $\Pi = (R, A, \mathbf{S})$.

Given user data $\vec{x} \in \mathcal{X}^n$ we will write $\Pi(\vec{x})$ to denote the distribution of the protocol's output when the users' data is $\vec{x}$, and $\vec{R}(\vec{x})$ denotes the distribution of the protocol's messages. Given a distribution $\mathbf{P}$ over $\mathcal{X}$, we will write $\Pi(\mathbf{P})$ and $\vec{R}(\mathbf{P})$ to denote the resulting distributions when $\vec{x}$ consists of n independent samples from $\mathbf{P}$.

Informally, we say that the protocol satisfies *local differential privacy* [1, 4, 5] if the local randomizers depend only very weakly on their inputs. Formally,

Definition II.1 (Local DP [1, 4, 5])**.** A protocol $\Pi = ((R_1, \dots, R_n), A, \mathbf{S})$ satisfies (ε, δ)*-local differential privacy* if for every $i \in [n]$, every $x, x' \in \mathcal{X}$, every $S \in \mathcal{S}$ and every $Y \subseteq \mathcal{Y}$,

$$\mathop{\mathbb{P}}_{R_i}[R_i(x, S) \in Y] \leq e^{\varepsilon} \cdot \mathop{\mathbb{P}}_{R_i}[R_i(x', S) \in Y] + \delta$$

where we stress that the randomness is *only* over the coins of R_i. If $\delta = 0$, we simply write ε*-local differential privacy*.

B. Threat Model: Manipulation Attacks

We capture manipulation attacks via a game involving a protocol $\Pi = (\vec{R}, A, \mathbf{S})$, a vector $\vec{x}$ of n data values, and an adversary M. We parameterize the game by the number of users n and the number of corrupted users $m \leq n$, written as $\mathrm{Manip}_{m,n}$; when clear from context, the subscript is omitted. The crux of the game is that the adversary corrupts a set C of at most m users, then the users are assigned data $\vec{x}$, and then either play *honestly* by sending the message $y_i = R_i(x_i, b)$ or they *manipulate* by playing some arbitrary message chosen by the adversary. Figure 3 presents the structure of an attack in the case where $C = \{1, 2\}$.

The game is described in Figure 4, including a possible restriction on the attacker. We use $\mathrm{Manip}_{m,n}(\Pi, \vec{x}, M)$ to denote the distribution on outputs of the protocol on data $\vec{x}$ and messages manipulated by M, and $\mathrm{Manip}_{m,n}(\vec{R}, \vec{x}, M)$

to denote the distribution of messages in the protocol. Given a distribution $\mathbf{P}$ over $\mathcal{X}$, we will use $\mathrm{Manip}_{m,n}(\Pi, \mathbf{P}, M)$ and $\mathrm{Manip}_{m,n}(\vec{R}, \mathbf{P}, M)$ to denote the resulting distributions when $\vec{x}$ consists of n independent samples from $\mathbf{P}$.

Parameters: $0 \leq m \leq n$.
Elements: A protocol $\Pi = (\vec{R}, A, \mathbf{S})$ for n users, a vector of data $\vec{x}$, an attacker M.

1) Each user i is given data x_i.
2) The public string $S \sim \mathbf{S}$ is sampled.
3) The attacker M chooses a set of *corrupted users* $C \subseteq [n]$ of size $\leq m$.

   If the corruptions are independent of the public string S then they are *public-string-oblivious*, and otherwise they are *public-string-adaptive*.
4) The attacker M chooses a set of messages $\{y_i\}_{i \in S}$ for the corrupted users.
5) The non-corrupted users $i \notin C$ choose messages $y_i \sim R_i(x_i, b)$ honestly.
6) The aggregator returns $z \leftarrow A(y_1, \ldots, y_n, b)$.

Fig. 4: Manipulation Game $\mathrm{Manip}_{m,n}$

C. Notational Conventions

Throughout, boldface roman letters indicate distributions (e.g. $\mathbf{P}$). Vectors are denoted $\vec{v} = (v_1, v_2, \ldots)$. We write $[n]$ to denote the set $\{1, \ldots, n\}$.

We use $\mathbf{Rad}(\mu)$ to denote the distribution over $\{\pm 1\}$ with mean μ, so that $\mathbb{P}[\mathbf{Rad}(\mu) = +1] = \frac{1+\mu}{2}$. Note that $\mathbf{Rad}(0)$ is uniform on $\{\pm 1\}$.

III. Attacks Against Protocols for Binary Data

In this section, we show how to attack any protocol that estimates the mean of a Rademacher distribution $\mathbf{Rad}(\mu)$. In particular, we show that any such protocol has error $\Omega(\frac{m}{\varepsilon n})$ in the presence of m corrupt users.

We begin with the result from [24] that decomposes any differentially private randomizer into a mixture of distributions:

Lemma III.1 (Adapted from [24])**.** *If $R : \{\pm 1\} \to \mathcal{Y}$ satisfies (ε, δ)-differential privacy, then there exist distributions $R^{(+1)}, R^{(-1)}, R^{\perp}, R^{\top}$ such that, for both $x = +1$ and $x = -1$,*

$$R(x) = \begin{cases} R^{(x)} & \textit{with probability } \frac{e^{\varepsilon}}{e^{\varepsilon}+1} \cdot (1-\delta) \\ R^{(-x)} & \textit{with probability } \frac{1}{e^{\varepsilon}+1} \cdot (1-\delta) \\ R^{\perp} & \textit{with probability } \delta \textit{ if } x = -1 \\ R^{\top} & \textit{with probability } \delta \textit{ if } x = +1 \end{cases}$$

The analysis of our attack will assume data is drawn from a distribution, so the following corollary will be useful:

Corollary III.2. *If $R : \{\pm 1\} \to \mathcal{Y}$ satisfies (ε, δ)-differential privacy, then there exist distributions $R^{(+1)}, R^{(-1)}$ such that, for all $\mu \in [-1, +1]$, $R(\mathbf{Rad}(\mu))$ is within statistical distance δ of the mixture $(\frac{1}{2} + \frac{e^{\varepsilon}-1}{e^{\varepsilon}+1} \cdot \frac{\mu}{2}) \cdot R^{(+1)} + (\frac{1}{2} - \frac{e^{\varepsilon}-1}{e^{\varepsilon}+1} \cdot \frac{\mu}{2}) \cdot R^{(-1)}$.*

Our attack, Algorithm 1 below, takes advantage of this structure of R by skewing the mixture ratio. Hence, no aggregator can tell if messages were generated from data with large mean or by manipulating the protocol.

Algorithm 1: A manipulation attack $M^{\vec{R}}_{m,n}$ against any protocol using n differentially private randomizers $\vec{R}$

```
For each $i \in [n]$:
  1) Add $i$ to $C$ with probability $m/2n$.
  2) If $|C| = m$ break the loop
For each corrupted user $i \in C$, report $y_i \sim R_i^{(+1)}$.
```

Lemma III.3. *For any $n > m > 18$ and any n randomizers $\vec{R}$ that satisfy (ε, δ) differential privacy, the distribution $\mathrm{Manip}_{m,n}(\vec{R}, \mathbf{Rad}(0), M^{\vec{R}}_{m,n})$ cannot be distinguished from $\vec{R}(\mathbf{Rad}(\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1} \cdot \frac{m}{2n}))$ with arbitrarily low probability of failure. Specifically, the statistical distance is at most $1/10 + 2n\delta$.*

Proof. In the first part of the proof we will argue that $M^{\vec{R}}_{m,n}$ behaves similarly to the alternative attack $\widetilde{M}^{\vec{R}}_{m,n}$ in which we eliminate step (2) of the for loop and choose whether or not to corrupt each user independently. Note that this attack will not always satisfy our budget of m corruptions, so it is not a valid attack in our model, but it is nonetheless useful for the analysis. The second part shows that $\mathrm{Manip}_{m,n}(\vec{R}, \mathbf{Rad}(0), \widetilde{M}^{\vec{R}}_{m,n})$ is approximately the same as having each user i independently sample from the mixture

$$\mathbf{P}_i := (\tfrac{1}{2} + \tfrac{m}{4n}) R_i^{(+1)} + (\tfrac{1}{2} - \tfrac{m}{4n}) R_i^{(-1)}.$$

The final part appeals to Corollary III.2 to prove that $\vec{\mathbf{P}}$ is well-approximated by $\vec{R}(\mathbf{Rad}(\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1} \cdot \frac{m}{2n}))$.

First, we claim that the statistical distance between

$$\mathrm{Manip}_{m,n}(\vec{R}, \mathbf{Rad}(0), M^{\vec{R}}_{m,n})$$

and

$$\mathrm{Manip}_{m,n}(\vec{R}, \mathbf{Rad}(0), \widetilde{M}^{\vec{R}}_{m,n})$$

is at most 1/10. These distributions only differ in the event that we hit $|C| = m$ and stop the loop early. This happens with probability exactly $\mathbb{P}[\mathbf{Bin}(n, m/2n) > m]$, and by standard bounds, this probability is at most $1/10$ whenever $m \geq 18$.

Next, we argue that the statistical distance between $\mathrm{Manip}_{m,n}(\vec{R}, \mathbf{Rad}(0), \widetilde{M}^{\vec{R}}_{m,n})$ and $\vec{\mathbf{P}}$ is at most $n\delta$. This is achieved by proving that the i-th user's message is sampled from a distribution within δ of $\mathbf{P}_i$. Note that

$$\mathbf{P}_i = \tfrac{m}{2n} \cdot R_i^{(+1)} + (1 - \tfrac{m}{2n})(\tfrac{1}{2} \cdot R_i^{(+1)} + \tfrac{1}{2} \cdot R_i^{(-1)}) \qquad (1)$$

Corruption status in $\widetilde{M}^{\vec{R}}_{m,n}$ is determined by a Bernoulli process with probability $\frac{m}{2n}$. If corrupted, user i will sample from $R_i^{(+1)}$; this corresponds to first term of (1). If not, Corollary III.2 implies that their message distribution $R_i(\mathbf{Rad}(0))$ is within δ of $\frac{1}{2} \cdot R_i^{(+1)} + \frac{1}{2} \cdot R_i^{(-1)}$; this corresponds to the second term of (1). Thus, the i-th distribution is within δ of (1). Since each of the n messages are independent, the overall difference between the distributions is at most $n\delta$.

Finally, Corollary III.2 implies that $\vec{R}(\mathbf{Rad}(\frac{e^\varepsilon+1}{e^\varepsilon-1} \cdot \frac{m}{2n}))$ is within statistical distance $n\delta$ of $\vec{\mathbf{P}}$. □

A consequence of Lemma III.3 is that no private protocol can estimate both $\mathbf{Rad}(0)$ and $\mathbf{Rad}(\frac{e^\varepsilon+1}{e^\varepsilon-1} \cdot \frac{m}{2n})$ with high accuracy under manipulation.

Theorem III.4. *For any $n > m > 18$ and any $\delta < 1/20n$, if $\Pi = (\vec{R}, A)$ is an (ε, δ) differentially private local protocol for n users and with probability $\geq 95/100$ it estimates $\mathbf{Rad}(\frac{e^\varepsilon+1}{e^\varepsilon-1} \cdot \frac{m}{2n})$ to within $\frac{e^\varepsilon+1}{e^\varepsilon-1} \cdot \frac{m}{4n}$, then with probability $\geq 3/4$ it does not estimate $\mathbf{Rad}(0)$ to within $\frac{e^\varepsilon+1}{e^\varepsilon-1} \cdot \frac{m}{4n}$ under attack $M^{\vec{R}}_{m,n}$.*

IV. Attacks Against Protocols for Large Data Universes

In this section, we show that more powerful manipulation attacks are possible when the data universe is $[d]$ for $d > 2$. For binary data, our attack showed that for any protocol there are two distributions $\mathbf{U}$ and $\mathbf{P}$ (i.e. $\mathbf{Rad}(0)$ and $\mathbf{Rad}(\mu(m, n, \varepsilon))$ with large statistical distance that are indistinguishable under manipulation. Specifically, $\|\mathbf{U} - \mathbf{P}\|_1 = \Omega(\frac{1}{\varepsilon\sqrt{n}} + \frac{m}{\varepsilon n})$ where $\|\mathbf{U} - \mathbf{P}\|_1$ denotes the ℓ_1 distance between the distributions $\sum_{j=1}^{d} |\mathbf{U}(j) - \mathbf{P}(j)|$. Here, we show that there is an attack and a distribution $\mathbf{P}$ such that $\|\mathbf{U} - \mathbf{P}\|_1 = \Omega\Big(\sqrt{\frac{d}{\log n}}(\frac{1}{\varepsilon\sqrt{n}} + \frac{m}{\varepsilon n})\Big)$ and $\mathbf{U}, \mathbf{P}$ are indistinguishable under this attack. This construction implies lower bounds for uniformity testing (given samples from $\mathbf{P}$, determine if $\mathbf{P} = \mathbf{U}$ or if $\|\mathbf{P} - \mathbf{U}\|_1$ is large) and ℓ_1 estimation (given samples from $\mathbf{P}$, report $\mathbf{P}'$ such that $\|\mathbf{P} - \mathbf{P}'\|_1$ is small).

A. A Family of Data Distributions

In this section, we show a particular way to convert a Rademacher distribution into a distribution over $[d]$. For a given partition of $[d]$ into $H, \overline{H}$ where $|H| = d/2$, we map the value $+1$ to a uniform element of H and -1 to a uniform element of $\overline{H}$. Thus, when $x \sim \mathbf{Rad}(\mu)$, we obtain a corresponding random variable $\hat{x}$ over $[d]$ whose distribution is $\mathbf{P}_{H,\mu}$ (see (2) below). Notice that estimating $\mathbb{P}[\hat{x} \in H]$ implies estimating μ.

$$\mathbf{P}_{H,\mu} := \begin{cases} \text{Uniform over } H \text{ with probability } \frac{1}{2} + \frac{\mu}{2} \\ \text{Uniform over } \overline{H} \text{ otherwise} \end{cases} \tag{2}$$

The algorithm $Q_{H,R}$ (Algorithm 2) performs the encoding of binary data $x \in \{\pm 1\}$ into $\hat{x} \in [d]$ then executes the randomizer R. Claim IV.1 is immediate from the construction.

Claim IV.1. *For any local randomizer $R : [d] \to \mathcal{Y}$, $H \subset [d]$ with size $d/2$, and $\mu \in [-1, +1]$, the execution of $Q_{H,R}$ (Algorithm 2) on a value drawn from $\mathbf{Rad}(\mu)$ is equivalent with the execution of R on a value drawn from $\mathbf{P}_{H,\mu}$:*

$$Q_{H,R}(\mathbf{Rad}(\mu)) = R(\mathbf{P}_{H,\mu})$$

```
Algorithm 2: $Q_{H,R}$ a local randomizer for binary data
  Parameters: A subset $H \subset [d]$ with size $d/2$; a local
    randomizer $R : [d] \to \mathcal{Y}$
  Input: $x \in \{\pm 1\}$

  If $x = 1$ then sample $\hat{x}$ uniformly from $H$
  Otherwise, sample $\hat{x}$ uniformly from $\overline{H}$.
  Return $y \sim R(\hat{x})$
```

Given a vector of n randomizers $\vec{R} = (R_1, \ldots, R_n)$, let $\vec{Q}_H$ denote the vector $(Q_{H,R_1}, \ldots, Q_{H,R_n})$. We can immediately generalize Claim IV.1 to multiple randomizers:

Claim IV.2. *For any n randomizers $\vec{R} = (R_1, \ldots, R_n)$ for data universe $[d]$, any $H \subset [d]$ with size $d/2$, and $\mu \in [-1, +1]$, the execution of $\vec{Q}_H$ on a sample from $\mathbf{Rad}(\mu)$ is equivalent with the execution of R on a sample from $\mathbf{P}_{H,\mu}$:*

$$\vec{Q}_H(\mathbf{Rad}(\mu)) = \vec{R}(\mathbf{P}_{H,\mu})$$

B. The Attack

In this subsection, we describe how to attack any differentially private protocol for d-ary data; to remove ambiguity with $M^{\vec{R}}_{m,n}$ (Algorithm 1), the attack will be denoted $M^{\vec{R}}_{d,m,n}$. As specified in Algorithm 3, the first step is to sample a uniformly random H. We show that if all $Q_{H,R_1}, \ldots, Q_{H,R_n}$ satisfy (ε, δ) differential privacy, then this attack inherits guarantees from the previous section. Then we show that the randomizers have strong privacy parameters with constant probability.

We begin the analysis of $M^{\vec{R}}_{d,m,n}$ by considering its behavior *conditioned on a fixed choice of H*. This restricted form will be denoted $M^{\vec{R},H}_{d,m,n}$. Then we analyze how the random choice of H gives the desired lower bound.

```
Algorithm 3: An attack $M^{\vec{R}}_{d,m,n}$ against any protocol using
  $n$ differentially private randomizers $\vec{R}$ for $d$-ary data
  Sample $H$ uniformly from all subsets of $[d]$ with size $d/2$
  For $i \in [n]$, add $i$ to $C$ with probability $m/2n$.
  If $|C| > m$, remove uniformly random members until
    $|C| = m$.
  For each corrupted user $i \in C$, report $y_i \sim Q^{(+1)}_{H,R_i}$
```

1) Analysis for fixed set H: Here, we show that manipulating $\vec{R}$ with $M^{\vec{R},H}_{d,m,n}$ induces the same distribution as if we had manipulated $\vec{Q}_H$ with $M^{\vec{Q}_H}_{m,n}$:

Claim IV.3. *Fix any n randomizers $\vec{R} = (R_1, \ldots, R_n)$ for data universe $[d]$, any $m \leq n$, and any $H \subset [d]$ with size $d/2$. If each Q_{H,R_i} is (ε, δ)-differentially private, then for any value $p \in [-1, +1]$, the distribution* $\mathrm{Manip}(\vec{R}, \mathbf{P}_{H,\mu}, M^{\vec{R},H}_{d,m,n})$ *is identical to* $\mathrm{Manip}\Big(\vec{Q}_H, \mathbf{Rad}(\mu), M^{\vec{Q}_H}_{m,n}\Big)$

Proof. To simplify the presentation, we assume that the users are sorted so that $C = \{1, \ldots, |C|\}$.

$$\begin{aligned}
&\mathrm{Manip}\Big(\vec{R}, \mathbf{P}_{H,\mu}, M^{\vec{R},H}_{d,m,n}\Big) \\
&= (Q^{(+1)}_{H,R_i})_{i \le |C|} \times (R_i(\mathbf{P}_{H,\mu}))_{i > |C|} \qquad \text{(By construction)} \\
&= (Q^{(+1)}_{H,R_i})_{i \le |C|} \times (Q_{H,R_i}(\mathbf{Rad}(\mu)))_{i > |C|} \qquad \text{(Claim IV.1)} \\
&= \mathrm{Manip}\Big(\vec{Q}_H, \mathbf{Rad}(\mu), M^{\vec{Q}_H}_{m,n}\Big)
\end{aligned}$$

The final equality follows from the fact that $|C|$ in $M^{\vec{R},H}_{d,m,n}$ is distributed identically with its counterpart in $M^{\vec{Q}_H}_{m,n}$. □

Claims IV.2 and IV.3 imply that we can use the analysis of $M^{\vec{R}}_{m,n}$ for our new attack $M^{\vec{R},H}_{d,m,n}$ *provided that* (ε, δ) privacy holds for all Q_{H,R_i}:

Lemma IV.4. *Fix any n randomizers $\vec{R}$, any $m \le n$, and any $H \subset [d]$ with size $d/2$. If each Q_{H,R_i} is (ε, δ)-differentially private, then there exists a value $\mu \in [-1, +1]$ such that the statistical distance between $\vec{R}(\mathbf{P}_{H,\mu})$ and $\mathrm{Manip}\Big(\vec{R}, \mathbf{U}, M^{\vec{R},H}_{m,n}\Big)$ is at most $1/10 + 2n\delta$ even though*

$$\|\mathbf{U} - \mathbf{P}_{H,\mu}\|_1 = \frac{e^\varepsilon + 1}{e^\varepsilon - 1} \cdot \frac{m}{2n} \tag{3}$$

Proof. By Claim IV.2, $\vec{Q}_H(\mathbf{Rad}(0)) = \vec{R}(\mathbf{P}_{H,0})$. Note that $\mathbf{P}_{H,0} = \mathbf{U}$. By Claim IV.3, $\mathrm{Manip}(\vec{R}, \mathbf{P}_{H,\mu}, M^{\vec{R},H}_{d,m,n})$ is identical to $\mathrm{Manip}\Big(\vec{Q}_H, \mathbf{Rad}(\mu), M^{\vec{Q}_H}_{m,n}\Big)$. So it will suffice to bound the statistical distance between $\vec{Q}_H(\mathbf{Rad}(0))$ and $\mathrm{Manip}\Big(\vec{Q}_H, \mathbf{Rad}(\mu), M^{\vec{Q}_H}_{m,n}\Big)$ for some choice of μ. But Lemma III.3 implies that for $\mu = \frac{e^\varepsilon+1}{e^\varepsilon-1} \cdot \frac{m}{2n}$, the distance is $1/10 + 2n\delta$.

It remains to prove (3). When sampling $x \sim \mathbf{P}_{H,\mu}$, the probability that $x = h$ is $\frac{1+\mu}{d}$ for each $h \in H$ and $\frac{1-\mu}{d}$ for each $h \notin H$. Hence,

$$\begin{aligned}
\|\mathbf{U} - \mathbf{P}_{H,\mu}\|_1 &= \frac{d}{2} \cdot \left|\frac{1}{d} - \frac{1+\mu}{d}\right| + \frac{d}{2} \cdot \left|\frac{1}{d} - \frac{1-\mu}{d}\right| \\
&= \mu = \frac{e^\varepsilon + 1}{e^\varepsilon - 1} \cdot \frac{m}{2n}
\end{aligned}$$

This concludes the proof. □

2) Analysis for randomized H: Here, we obtain a lower bound by analyzing randomness in H. For clarity of exposition, the analysis in this section is limited to pure differential privacy. In the full version of this work, we prove more general statements that encompass approximate differential privacy.

We begin with a lemma that bounds the privacy parameter of all Q_{H,R_i} by an ε' that depends on $\vec{R}$: we will use $|\vec{R}|_{\neq}$ to denote the number of unique randomizers in $\vec{R}$.

Lemma IV.5. *Fix any $\vec{R}$ where each $R_i : [d] \to \mathcal{Y}$ is ε differentially private. There is a constant c such that, if $d > c \cdot (e^\varepsilon - 1)^2 \ln\Big(|\mathcal{Y}| \cdot |\vec{R}|_{\neq}\Big)$ and H is drawn uniformly from all subsets of $[d]$ with size $d/2$, then the following holds with probability $> 2/3$ over the randomness of H: Every Q_{H,R_i} specified by Algorithm 2 is ε' differentially private, where*

$$\varepsilon' = (e^\varepsilon - 1)\sqrt{\frac{c}{d} \ln\Big(|\mathcal{Y}| \cdot |\vec{R}|_{\neq}\Big)}$$

We continue with a bound that only depends on n and not any particular structure in $\vec{R}$:

Lemma IV.6. *Fix any $\vec{R}$ where each $R_i : [d] \to \mathcal{Y}$ is ε differentially private. There is a constant c such that, if $d > c \cdot (e^\varepsilon - 1)^2 \ln(e^\varepsilon n)$ and H is drawn uniformly from all subsets of $[d]$ with size $d/2$, then the following holds with probability $> 2/3$ over the randomness of H: Every Q_{H,R_i} specified by Algorithm 2 is $(\varepsilon', 1/180n)$ differentially private, where*

$$\varepsilon' = (e^\varepsilon - 1)\sqrt{\frac{c}{d} \ln(e^\varepsilon n)}$$

Proofs of these statements can be found in Appendix A. From Lemmas IV.4, IV.5, and IV.6, the attack $M^{\vec{R}}_{d,m,n}$ successfully obscures a uniform distribution with probability 2/3:

Lemma IV.7. *Fix any $n > m > 18$, any $\varepsilon < 1$, and any ε-locally private protocol $\Pi = (\vec{R}, A)$ that accepts data from $[d]$. There are constants c_0, c_1 and a value $\mu \in [-1, +1]$ such that the following holds: if $d > c_0 \cdot (e^\varepsilon - 1)^2 \ln\Big(\min n, |\mathcal{Y}| \cdot |\vec{R}|_{\neq}\Big)$ then, with probability $> 2/3$, $M^{\vec{R}}_{d,m,n}$ chooses H such that the statistical distance between $\vec{R}(\mathbf{P}_{H,\mu})$ and $\mathrm{Manip}\Big(\vec{R}, \mathbf{U}, M^{\vec{R},H}_{d,m,n}\Big)$ is at most $1/9$ even though*

$$\|\mathbf{U} - \mathbf{P}_{H,\mu}\|_1 \ge \frac{c_1 \cdot m\sqrt{d}}{\varepsilon n \sqrt{\ln\Big(\min n, |\mathcal{Y}| \cdot |\vec{R}|_{\neq}\Big)}}$$

C. Applications to Testing and Estimation

From Lemma IV.7, we obtain lower bounds on how well the manipulation attack fares against protocols for uniformity testing and estimation.

Theorem IV.8. *Fix any $n > m > 18$, any $\varepsilon < 1$, and any ε-locally private protocol $\Pi = (\vec{R}, A)$ for testing uniformity over $[d]$. There are constants c_0, c_1 such that for all $d > c_0 \cdot (e^\varepsilon - 1)^2 \ln\Big(\min n, |\mathcal{Y}| \cdot |\vec{R}|_{\neq}\Big)$ if $\mathbb{P}[\Pi(\mathbf{P}) = \text{“not uniform”}] \ge 95/100$ for all*

$$\|\mathbf{U} - \mathbf{P}\|_1 \ge \frac{c_1 \cdot m\sqrt{d}}{\varepsilon n \sqrt{\ln\Big(\min n, |\mathcal{Y}| \cdot |\vec{R}|_{\neq}\Big)}} \tag{4}$$

then $\mathbb{P}\Big[\mathrm{Manip}\Big(\Pi, \mathbf{U}, M^{\vec{R}}_{d,m,n}\Big) = \text{“not uniform”}\Big] > 1/2$

Theorem IV.9. *Fix any $n > m > 18$, any $\varepsilon < 1$, and any ε-private protocol for estimating distributions over $[d]$. There exists constants c_0, c_1 such that, for all $d > c_0 \cdot (e^\varepsilon - 1)^2 \ln(\min n, |\mathcal{Y}||\vec{R}|_{\neq})$ and $\alpha = \frac{c_1 \cdot m\sqrt{d}}{\varepsilon n \sqrt{\ln\Big(\min n, |\mathcal{Y}| \cdot |\vec{R}|_{\neq}\Big)}}$ if $\mathbb{P}[\|\Pi(\mathbf{P}) - \mathbf{P}\|_1 < \alpha] > 95/100$ for all distributions $\mathbf{P}$, then $\mathbb{P}\Big[\Big\|\mathrm{Manip}\Big(\Pi, \mathbf{U}, M^{\vec{R}}_{d,m,n}\Big) - \mathbf{U}\Big\|_1 \ge \alpha\Big] > 1/2$.*

V. Protocols with Nearly Optimal Robustness to Manipulation

In this section, we consider a number of well-studied problems in local privacy and identify specific protocols from the literature with optimal robustness to manipulation (i.e. matching the lower bounds implied by our attacks). As discussed in the introduction, most of these problems can be cast as accurately mean estimation of bounded vectors. We also give an additional protocol for computing heavy hitters in the Appendix.

A. Warmup: Mean Estimation for Binary Data

As a warmup, we analyze the randomized response protocol in the presence of manipulation. The protocol is defined by the local randomizer $R^{\mathrm{RR}}_{\varepsilon}$ and aggregator $A^{\mathrm{RR}}_{n,\varepsilon}$ as follows:

$$R^{\mathrm{RR}}_{\varepsilon}(x) := \begin{cases} \frac{e^{\varepsilon}+1}{e^{\varepsilon}-1}\cdot x & \text{with probability } \frac{e^{\varepsilon}}{e^{\varepsilon}+1} \\ -\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1}\cdot x & \text{with probability } \frac{1}{e^{\varepsilon}+1} \end{cases}$$
$$A^{\mathrm{RR}}_{n,\varepsilon}(\vec{y}) := \tfrac{1}{n}\textstyle\sum_{i=1}^{n} y_i$$

We bound the error of this protocol by $O(\frac{1}{\varepsilon}(\frac{1}{\sqrt{n}}+\frac{m}{n})$, which matches the lower bound of Theorem III.4 up to constants.

Theorem V.1. *For any positive integers $m \le n$, any $\varepsilon > 0$, any $\vec{x} \in \{0,1\}^n$, any manipulation adversary M, and any $\beta > 0$, with probability $\ge 1-\beta$, we have*

$$\left|\mathrm{Manip}_{m,n}(RR_{\varepsilon,n}, \vec{x}, M) - \tfrac{1}{n}\textstyle\sum_{i=1}^{n} x_i\right|$$
$$< \tfrac{e^{\varepsilon}+1}{e^{\varepsilon}-1}\cdot\left(\sqrt{\tfrac{2}{n}\ln\tfrac{2}{\beta}} + \tfrac{2m}{n}\right)$$

Proof. Consider an execution of $\mathrm{Manip}(\mathtt{RR}_{\varepsilon,n}, \vec{x}, M)$. Let C be the set of corrupted users, let $y_1, \dots, y_n$ be the messages sent in the protocol and let $\underline{\vec{y}}$ be the messages that would have been sent in an honest execution (so $\underline{y}_i = y_i$ for every $i \notin C$). Let $z = \frac{1}{n}\sum_{i=1}^{n} y_i$ be the output of the aggregator.

We can break up the error into two components, one corresponding to the error of the honest execution and one corresponding to the error introduced by manipulation.

$$\begin{aligned}
&\left|\tfrac{1}{n}\textstyle\sum_{i\in[n]} y_i - \tfrac{1}{n}\sum_{i\in[n]} x_i\right| \\
&= \left|\tfrac{1}{n}\textstyle\sum_{i\in[n]} y_i - \tfrac{1}{n}\sum_{i\in[n]} \underline{y}_i + \tfrac{1}{n}\sum_{i\in[n]} \underline{y}_i - \tfrac{1}{n}\sum_{i\in[n]} x_i\right| \\
&\le \left|\tfrac{1}{n}\textstyle\sum_{i\in[n]} y_i - \tfrac{1}{n}\sum_{i\in[n]} \underline{y}_i\right| + \left|\tfrac{1}{n}\sum_{i\in[n]} \underline{y}_i - \tfrac{1}{n}\sum_{i\in[n]} x_i\right| \\
&= \underbrace{\left|\tfrac{1}{n}\textstyle\sum_{i\in C} y_i - \underline{y}_i\right|}_{\text{manipulation}} + \underbrace{\left|\tfrac{1}{n}\textstyle\sum_{i\in[n]} \underline{y}_i - \tfrac{1}{n}\sum_{i\in[n]} x_i\right|}_{\text{honest execution}}
\end{aligned}$$

Since each message in the protocol is either $\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1}$ or $-\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1}$, we have $|y_i - \underline{y}_i| \le 2\cdot\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1}$. Thus, the manipulation term is bounded by $\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1}\cdot\frac{2m}{n}$ with probability 1.

For the error of the honest execution, note that $\mathbb{E}[\underline{y}_i] = x_i$ and $\frac{1}{n}\sum_{i\in[n]}\underline{y}_i$ is an average of n independent random variables bounded to a range of width $2\cdot\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1}$. Thus, by Hoeffding's inequality, the second term is bounded by $\frac{e^{\varepsilon}+1}{e^{\varepsilon}-1}\sqrt{\frac{2\ln(2/\beta)}{n}}$ with probability at least $1-\beta$. □

Our analysis of richer protocols has the same structure. We construct the protocol so that each message y_i gives an unbiased estimate of x_i, and the aggregation computes the mean of the messages. We then isolate the effect of the manipulation from that of an honest execution. Finally, we bound the influence of m messages on the output of the protocol. For richer protocols the analysis of the final step will become more involved.

B. Mean Estimation

We consider vector-valued data in $\mathbb{R}^d$. For any $p \ge 1$, $\|x\|_p := (\sum_{j=1}^{d}|x_j|^p)^{1/p}$ denotes the standard ℓ_p norm and B^d_p denotes the ℓ_p unit ball in $\mathbb{R}^d$. As is standard $\|x\|_\infty = \max_{j\in[d]}|x_j|$ is the ℓ_∞ norm and B^d_∞ is the ℓ_∞ unit ball. In this section, we study instances of the general ℓ_p/ℓ_q *mean estimation problem*: given data $x_1, \dots, x_n \in B^d_p$, output some $\hat{\mu}$ such that $\left\|\hat{\mu} - \frac{1}{n}\sum_i x_i\right\|_q$ is as small as possible.

ℓ_∞/ℓ_∞ estimation (Counting Queries). In this problem, each user has data $x_i \in B^d_\infty$ and the goal is to obtain a vector $\hat{\mu}$ such that $\left\|\hat{\mu} - \frac{1}{n}\sum x_i\right\|_\infty$ is as small as possible. We consider the following protocol $\mathtt{EST\infty} = (R^{\mathtt{EST\infty}}, n, A^{\mathtt{EST\infty}})$, which is known to have optimal error absent manipulation.

1) Using public randomness, we partition users into d groups each of size n/d. Intuitively, we are assigning each group to one coordinate.
2) For each group j, each user i in group j reports the message $y_i \leftarrow R^{\mathrm{RR}}(x_{i,j})$
3) For each group j, the aggregator computes the average of the messages from group j to obtain $\hat{\mu}_j \approx \frac{1}{n}\sum_i x_{i,j}$. The aggregator reports $\hat{\mu} = (\hat{\mu}_1, \dots, \hat{\mu}_d)$

If the adversary's corruptions are oblivious to the public partition, then we show that there are $\approx m/d$ corrupt users in each group of size n/d. By our analysis of randomized response, the adversary can introduce at most $\approx \frac{m/d}{\varepsilon n/d} = \frac{m}{\varepsilon n}$ error in any single coordinate.

Theorem V.2. *For any $\varepsilon \in (0,1)$, any positive integers $m \le n$, any $x_1, \dots, x_n \in B^d_\infty$, and any* public-string-oblivious *adversary M, with probability $\ge 99/100$, the error $\left\|\mathrm{Manip}_{m,n}(EST\infty_{\varepsilon}, \vec{x}, M) - \frac{1}{n}\sum_{i=1}^{n} x_i\right\|_\infty$ is bounded by $O(\sqrt{\frac{d\log d}{\varepsilon^2 n}} + \frac{m}{\varepsilon n})$*

Observe that the dependence on m matches that of the lower bound in Theorem III.4 for Bernoulli estimation. We give the complete details of the protocol in the full version.

ℓ_1/ℓ_∞ Estimation (Histograms). In this problem, each user i has data $x_i \in B^d_1$ and the objective is a $\hat{\mu}$ such that $\left\|\hat{\mu} - \frac{1}{n}\sum_{i=1}^{n} x_i\right\|_\infty$ is as small as possible. To simplify the discussion, we focus on the special case where user i has data $x_i \in [d]$. Define $\mathit{freq}(j, \vec{x}) := \frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_{\{x_i = j\}}$ and $\mathit{freq}(\vec{x}) := (\mathit{freq}(1, \vec{x}), \dots, \mathit{freq}(1, \vec{x}))$. The objective is a vector $\hat{\mu}$ such that $\|\hat{\mu} - \mathit{freq}(\vec{x})\|_\infty$ is as small as possible.

We consider the following protocol $\mathtt{HST}_{\varepsilon}$, which is known to have optimal error absent manipulation:

1) For each user i, independently sample a uniform public vector $\vec{s}_i \in \{\pm 1\}^d$.

2) Each user i reports the message $y_i \leftarrow R^{\mathrm{RR}}_\varepsilon(s_{i,x_i})$.

3) The aggregator receives messages $y_1, \dots, y_n$ and outputs $\hat{\mu} \leftarrow \frac{1}{n}\sum_{i=1}^n y_i \cdot \vec{s}_i$.

Theorem V.3. *For any $\varepsilon \in (0,1)$, any positive integers $m \leq n$, any $x_1, \dots, x_n \in [d]$, and any adversary M, with probability at least $99/100$, we have*

$$\left\|\mathrm{Manip}_{m,n}(\mathtt{HST}_\varepsilon, \vec{x}, M) - \mathit{freq}(\vec{x})\right\|_\infty = O\left(\sqrt{\frac{\log d}{\varepsilon^2 n}} + \frac{m}{\varepsilon n}\right)$$

Proof Sketch. Identically to the proof of Theorem V.1, we partition the error contributed by the honest and corrupt users. Let $\vec{y}$ be the messages sent in the protocol and let $\underline{\vec{y}}$ be the messages that would have been sent in an honest execution. Below, $\sum_i \cdots$ will be short for $\sum_{i=1}^n \cdots$. We can write

$$\begin{aligned}&\left\|\tfrac{1}{n}\textstyle\sum_i \vec{s}_i - \mathit{freq}(\vec{x})\right\|_\infty \\ &= \underbrace{\max_{j\in[d]}\left[\tfrac{1}{n}\textstyle\sum_{i\in C}(y_i - \underline{y}_i)s_{i,j}\right]}_{\text{manipulation}} + \underbrace{\max_{j\in[d]}\left|\tfrac{1}{n}\textstyle\sum_i \underline{y}_i s_{i,j} - \mathbb{1}_{\{x_i=j\}}\right|}_{\text{honest execution}}\end{aligned}$$

To bound the error from the manipulation, note that messages have magnitude $\frac{e^\varepsilon+1}{e^\varepsilon-1} = \Theta(1/\varepsilon)$. Hence, the bias introduced to any coordinate j is at most $O(m/\varepsilon n)$ with probability 1.

We now bound the error of the honest execution. If $x_i = j$, the expectation of $\underline{y}_i s_{i,j}$ is 1. Otherwise, the expectation is 0 because of pairwise independence. Hence, the honest execution has 0 expected error. Because messages have magnitude $\Theta(1/\varepsilon)$, Hoeffding's inequality and a union bound imply that no frequency estimate is more than $O(\sqrt{\log d/\varepsilon^2 n})$ from $\mathit{freq}(j, \vec{x})$ with probability $\geq 99/100$. □

A slightly more general protocol can be used to obtain the same result for ℓ_1/ℓ_∞ estimation.

Theorem V.4. *For any $\varepsilon \in (0,1)$, there is an ε-locally private protocol $\mathtt{EST1}_\varepsilon$ such that for any positive integer n, any $x_1, \dots, x_n \in B^d_1$, and any adversary M, with probability $\geq 99/100$, the error $\left\|\mathrm{Manip}_{m,n}(\mathtt{EST1}_\varepsilon, \vec{x}, M) - \frac{1}{n}\sum_{i=1}^n x_i\right\|_\infty$ is bounded by $O(\sqrt{\frac{\log d}{\varepsilon^2 n}} + \frac{m}{\varepsilon n})$.*

We give the complete details of $\mathtt{EST1}_\varepsilon$ in the full version of the paper. Observe that its manipulation error matches that of Bernoulli estimation (Theorem III.4).

ℓ_1/ℓ_1 **Estimation (Frequency Estimation).** In this problem, each user i has data $x_i \in B^d_1$ and the objective is a $\hat{\mu}$ such that $\left\|\hat{\mu} - \frac{1}{n}\sum_{i=1}^n x_i\right\|_1$ is as small as possible. Because this problem and the ℓ_1/ℓ_∞ problem have the same data type, we consider the same protocols but change the analysis to upper bound ℓ_1 error.

Theorem V.5. *For any $\varepsilon \in (0,1)$, any positive integer n, any $x_1, \dots, x_n \in [d]$, and any adversary M, with probability at least $99/100$, the error $\left\|\mathrm{Manip}_{m,n}(\mathtt{HST}_\varepsilon, \vec{x}, M) - \frac{1}{n}\sum_{i=1}^n x_i\right\|_1$ is bounded by $\tilde{O}(\sqrt{\frac{d^2}{\varepsilon^2 n}} + \frac{m\sqrt{d}}{\varepsilon n})$.*

Proof Sketch. Identically to the proof of Theorem V.1, we partition the error contributed by the honest and corrupt users. Let $\vec{y}$ be the messages sent in the protocol and let $\underline{\vec{y}}$ be the messages that would have been sent in an honest execution. Let $S \in \{\pm 1\}^{d\times n}$ be the matrix whose columns are $\vec{s}_1, \dots, \vec{s}_n$, and $S_C \in \{\pm 1\}^{d\times |C|}$ be the submatrix consisting only columns corresponding to users $i \in C$. Then we can write

$$\begin{aligned}&\left\|\tfrac{1}{n}\textstyle\sum_{i=1}^n y_i \vec{s}_i - \mathit{freq}(\vec{x})\right\|_1 \\ &= \underbrace{\left\|\tfrac{1}{n}S_C(\vec{y}_C - \underline{\vec{y}}_C)\right\|_1}_{\text{manipulation}} + \underbrace{\sum_{j\in[d]}\left|\tfrac{1}{n}\textstyle\sum_{i=1}^n \underline{y}_i s_{i,j} - \mathbb{1}_{\{x_i=j\}}\right|}_{\text{honest execution}}\end{aligned}$$

To bound the error from the honest execution, observe that the expectation and variance are $O(\sqrt{1/\varepsilon^2 n})$ and $O(1/\varepsilon n)$, respectively, for any term in the outer sum. Hence, error has magnitude $O(\sqrt{d^2/\varepsilon^2 n})$ with probability $\geq 199/200$.

To bound the error from the manipulation, we will use bounds on the singular values of the random matrix S_C. As a shorthand, let $c_\varepsilon = \frac{e^\varepsilon+1}{e^\varepsilon-1}$. A calculation then shows

$$\left\|\tfrac{1}{n}S_C(\vec{y}_C - \underline{\vec{y}}_C)\right\|_1 \leq \frac{c_\varepsilon\sqrt{md}}{n} \max_{\substack{C\subseteq[n]\\|C|=m}} \|S_C\|_2$$

where $\|S_C\|_2$ denotes the largest singular value (operator norm) of S_C. Since each matrix $S_C \in \{\pm 1\}^{d\times m}$ is uniformly random, we can use bounds on the singular values of random matrices.

Lemma V.6 **(see e.g. the textbook [28]).** *For any $k \in \mathbb{R}_+$ larger than an absolute constant and a matrix $S_C \in \mathbb{R}^{d\times m}$ whose entries are sampled independently and identically, the following holds with probability $\geq 1 - \exp(-k(d+m))$ over the randomness of S_C.*

$$\|S_C\|_2 = O(\sqrt{kd} + \sqrt{km})$$

The adversary has $\binom{n}{m} \leq \exp(m \ln n)$ choices of corruptions C. By a union bound over that set, we have with probability at least $1 - \exp(m\ln n - k(m+d))$

$$\begin{aligned}\left\|\tfrac{1}{n}S_C(\vec{y}_C - \underline{\vec{y}}_C)\right\|_1 &\leq \frac{c_\varepsilon\sqrt{md}}{n}\cdot O(\sqrt{kd} + \sqrt{km}) \\ &= O\left(\sqrt{\frac{d^2 k}{\varepsilon^2 n}} + \frac{m\sqrt{dk}}{\varepsilon n}\right)\end{aligned}$$

For $k = O(\ln n)$, the bound holds with probability at least $199/200$. □

A slightly more general protocol can be used to obtain the same result for ℓ_1/ℓ_1 estimation.

Theorem V.7. *For any $\varepsilon \in (0,1)$, there is an ε-locally private protocol $\mathtt{EST1}$ such that for any positive integer n, any $x_1, \dots, x_n \in (B^d_1)^n$, and any adversary M, with probability $\geq 99/100$, the error $\left\|\mathrm{Manip}_{m,n}(\mathtt{EST1}_\varepsilon, \vec{x}, M) - \frac{1}{n}\sum_{i=1}^n x_i\right\|_1$ is bounded by $\tilde{O}(\sqrt{\frac{d^2}{\varepsilon^2 n}} + \frac{m\sqrt{d}}{\varepsilon n})$.*

Observe that the manipulation error matches the lower bound in Theorem IV.9, up to a logarithmic factor.

ℓ_2/ℓ_2 **Estimation.** In this problem, each user i has data $x_i \in B_2^d$ and the objective is a $\hat{\mu}$ such that $\left\|\hat{\mu} - \frac{1}{n}\sum_{i=1}^n x_i\right\|_2$ is as small as possible.

Consider the following protocol `EST2` adapted from [17]:

1) For each user i, we sample $\vec{s}_i \in \mathbb{R}^d$ uniformly at random from the surface of B_2^d.
2) Each user i computes $w_i \leftarrow \operatorname{sgn}(\vec{s}_i \cdot x_i)$ and then reports $y_i \leftarrow R_\varepsilon^{\mathrm{RR}}(w_i)$ to the aggregator
3) The aggregator receives the messages $y_1, \ldots, y_n$ and outputs $\vec{z} \leftarrow \frac{c\sqrt{d}}{n}\sum_{i=1}^n y_i \vec{s}_i$ for some constant $c > 0$.

Theorem V.8. *For any $\varepsilon \in (0,1)$, any positive integer n, any $x_1, \ldots, x_n \in B_2^d$, and any adversary M, with probability $\geq 99/100$, we have*

$$\left\|\mathrm{Manip}_{m,n}(\mathit{EST2}_\varepsilon, \vec{x}, M) - \frac{1}{n}\sum_{i=1}^n x_i\right\|_2 = \tilde{O}\left(\sqrt{\frac{d}{\varepsilon^2 n}} + \frac{m}{\varepsilon n}\right)$$

Observe that the manipulation error matches that of Bernoulli estimation (Theorem III.4) up to a logarithmic factor. The proof follows the same arc as before: we partition the error contributed by honest and corrupt users and use bounds on the singular values of the random matrix S_C to bound the manipulation error. Due to this repeated structure, we defer the proof to the Appendix.

C. Uniformity Testing

In this problem, each user has data $x_i \in [d]$ sampled from a distribution $\mathbf{P}$. If $\mathbf{P} = \mathbf{U}$, then a protocol for this problem should output "uniform" with probability $\geq 99/100$. If $\|\mathbf{P} - \mathbf{U}\|_1 > \alpha$, then it should output "not uniform" with probability $\geq 99/100$. Smaller values of α are desirable.

We consider the `RAPTOR` protocol, introduced by [15]. It divides users into G groups each of size n/G (where G is a parameter). In each group g,

1) Sample public set $S \in \{S \subset [d] \mid |S| = d/2\}$ uniformly at random.
2) Each user assigns $x_i' \leftarrow +1$ if $x_i \in S$ and otherwise $x_i' \leftarrow -1$
3) Each user i reports $y_i \leftarrow R_\varepsilon^{\mathrm{RR}}(x_i')$ to the aggregator
4) The aggregator computes the average of the messages: $\hat{\mu}_g \leftarrow \frac{G}{n}\sum y_i$.

If there is some $\hat{\mu}_g \gtrapprox \sqrt{\frac{1}{\varepsilon^2 n}} + \frac{m}{\varepsilon n}$, the aggregator reports "not uniform." Otherwise, it reports "uniform."

Theorem V.9. *There is a choice of parameter G such that, for any $\varepsilon \in (0,1)$, any positive integers $m \leq n$, and any adversary M, the following holds with probability $\geq 99/100$*

$$\mathrm{Manip}_{m,n}(\mathit{RAPTOR}_\varepsilon, \mathbf{U}, M) = \text{“uniform”}$$

and, when $\|\mathbf{P} - \mathbf{U}\|_1 \geq \alpha$ for some $\alpha = O\left(\sqrt{\frac{d}{\varepsilon^2 n}} + \frac{m\sqrt{d}}{\varepsilon n}\right)$, the following also holds with probability $\geq 99/100$

$$\mathrm{Manip}_{m,n}(\mathit{RAPTOR}_\varepsilon, \mathbf{P}, M) = \text{“not uniform”}$$

Proof Sketch. Consider any $g \in [G]$. When $\|\mathbf{P} - \mathbf{U}\|_1 \geq \sqrt{10d} \cdot \alpha$, a lemma by [15] implies that, with at least some constant probability over the randomness of S, $\left|\underset{x \sim \mathbf{P}}{\mathbb{P}}[x \in S] - \frac{1}{2}\right| \gtrapprox \alpha$. For $\alpha \gtrapprox \sqrt{G/\varepsilon^2 n} + mG/\varepsilon n$, RR_ε will provide an estimate of $\underset{x \sim \mathbf{P}}{\mathbb{P}}[x \in S]$ that is larger than $\frac{1}{2} + \alpha/2$. But when $\mathbf{P} = \mathbf{U}$, the protocol will give an estimate of $\underset{x \sim \mathbf{P}}{\mathbb{P}}[x \in S]$ that is less than $\frac{1}{2} + \alpha/2$. This means there is a threshold test that has a constant probability of succeeding. The G repetitions serve to increase the success probability to $99/100$. This completes the proof. □

Observe that the bound $\alpha = O(\frac{m\sqrt{d}}{\varepsilon n} + \sqrt{\frac{d}{\varepsilon^2 n}})$ matches the lower bound of Theorem IV.8 up to logarithmic factors. We give the complete details in the full version of the paper.

VI. Experiments

In this section we give a basic set of experiments with our attack against the natural frequency estimation protocol `HST`, which we showed to be optimally robust to manipulation (Theorem IV.9). These experiments validate our theoretical analysis by showing that—at least for the protocol `HST`—the vulnerability to manipulation depends significantly on the dimension of the input domain. The experiments also indicate that the concrete error introduced by the attack against the natural protocol `HST` is significantly larger than what our worst-case analysis guarantees against arbitrary protocols.

In our experiments, we generate data from the uniform distribution over the domain $\{1, \ldots, d\}$ and measure the ℓ_1 error of the protocol `HST`. In our experiments, we fix $n = 2 \times 10^5$ and $\varepsilon = 1.0$, and vary the dimension d and the fraction of corrupted users m/n. In Figure 5, we plot the median ℓ_1 error as well as the upper and lower quartiles of the error. Table II gives the approximate breakdown point for varying choices of d. For purposes of concreteness, we define the breakdown point as the fraction of corrupted users at which the error becomes at least 0.5, although we note that even much smaller error is likely unacceptable in applications.

We also do the same set of experiments with an alternative protocol `NR-HST` (for *non-robust* `HST`). This protocol differs from `HST` only in that the users samples a uniform vector $\vec{s}_i \in \{\pm 1\}^d$ themselves, and then sends $y_i \cdot \vec{s}_i$. For comparison, in `HST`, the user receives the vector $\vec{s}_i$ as public randomness, and only sends the single bit y_i. Note that if all users play honestly, then the distribution of the aggregator's output is *identical* to `HST`. However, since the corrupted users can now change how they choose $\vec{s}_i$ in addition to how they choose y_i, the protocol is much less robust to manipulation, and our experiments in Table 6 show that the protocol is much less robust to our attack.

Our final round of experiments reveal that, under a different measure of error, `NR-HST` is vulnerable to just a *small number* of corrupt users. The ℓ_1 norm scales with the quantity $\max_{S \subset [d]} |\sum_{j \in S} z_j - \mathit{freq}(j, \vec{x})|$, the *maximum* total error of any subset. But a data analyst may have little interest in the maximum and instead have a *target* subset, like frequencies of specific words. In Figure 7, we depict the total error of `NR-HST` on $S = \{1, \ldots, d/2\}$ for $n = 5 \cdot 10^4$ users. When

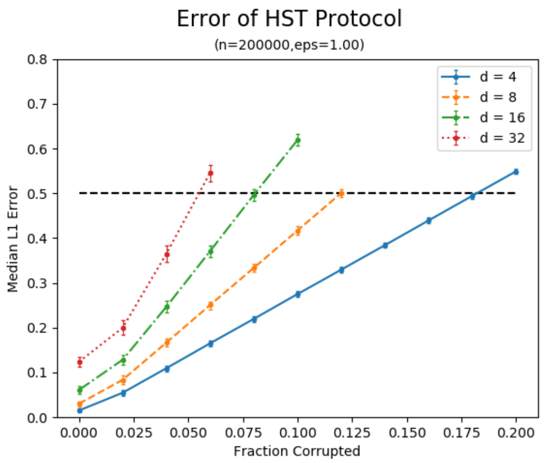

Fig. 5: ℓ_1-error of the HST protocol for $n = 2 \times 10^5$ users, $\varepsilon = 1.0$, and various choices of dimension d and the fraction of corrupted users m/n. Each point represents the median error across 896 trials. The bars depict the 25% and 75% quantiles. The horizontal line represents the breakdown point (error 0.5).

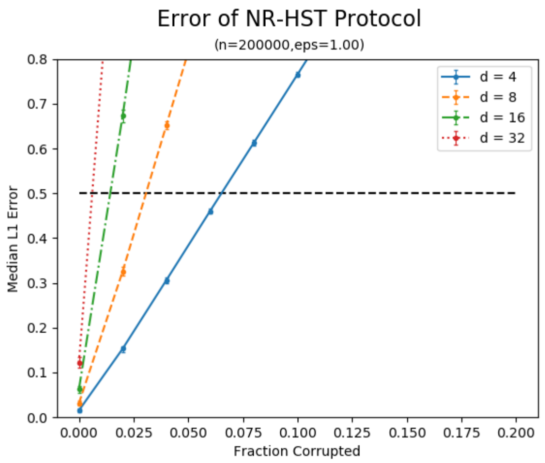

Fig. 6: ℓ_1 error of the NR-HST protocol for the same parameters as in Table II.

$d = 32$, this error is under 0.05 when there are no corrupted users but it increases by around a factor of 3 when there are *only 250 corrupted users*.

VII. Conclusion

This paper systematically studies *manipulation attacks* on locally differentially private protocols, in which malicious clients inject improperly generated messages into the protocol in order to influence its output. We show that vulnerability to such attacks is inherent to the model—-every noninteractive local protocol admits such attacks, and the attacks' effectiveness increases as the privacy guarantee gets stronger and, for some tasks, as the dimension of the data grows.

Dimension (d)	Breakdown Point (Error = 0.5)	
	HST	NR-HST
4	$\approx 18\%$	$\approx 7\%$
8	$\approx 12\%$	$\approx 3\%$
16	$\approx 8\%$	$< 2\%$
32	$\approx 5\%$	$\ll 1\%$

TABLE II: Upper bounds on the breakdown point (error 0.5) of the HST protocol for $n = 2 \times 10^5$, $\varepsilon = 1.0$, and various choices of dimension d.

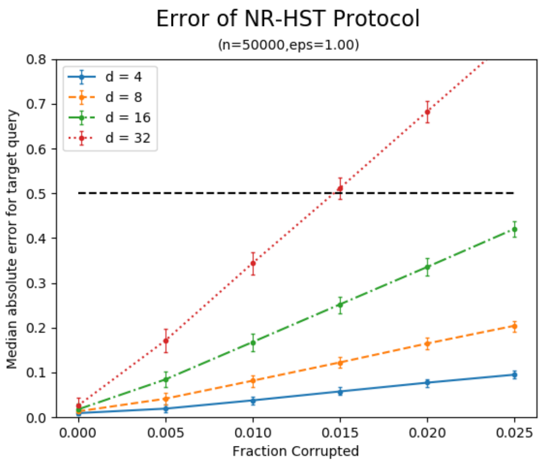

Fig. 7: Error of the NR-HST protocol for $n = 5 \cdot 10^4$ users, $\varepsilon = 1.0$, and various choices of d and m/n. Here, error is computed by taking the sum of frequency estimation errors of $\{1, \ldots d/2\}$.

Our work leaves open a number of technical questions. Can interactive local protocols resist manipulation more effectively than non-interactive protocols? Can we close the few remaining gaps between upper and lower bounds in Table I? More fundamentally, it highlights the importance of systems that collect and analyze sensitive information at scale with minimal trust requirements and strong privacy guarantees. Multiparty computation (as in [8] [11] and work on the shuffled model [12, 13, 27]) offers one set of possible solutions, and other effective alternatives surely remain to be found.

Acknowledgment

Part of this work was done while the authors were visiting the Simons Institute for Theory of Computing. The authors are grateful to Henry Corrigan-Gibbs, Úlfar Erlingsson, Vitaly Feldman, Gautam Kamath, Seth Neel, and Aaron Roth for helpful discussions, as well as to the anonymous reviewers from this conference for constructive critical feedback. The authors thank Jack Doerner for help preparing the figures.

REFERENCES

[1] A. Evfimievski, J. Gehrke, and R. Srikant, "Limiting privacy breaches in privacy preserving data mining," in *Proceedings of the 22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, ser. PODS '03. New York, NY, USA: ACM, 2003, pp. 211–222.

[2] Ú. Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized aggregatable privacy-preserving ordinal response," in *Proceedings of the ACM Conference on Computer Security*, ser. CCS'14. ACM, 2014, pp. 1054–1067.

[3] Apple Differential Privacy Team, "Learning with privacy at scale," December 2017.

[4] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proceedings of the 3rd Conference on Theory of Cryptography*, ser. TCC '06. Berlin, Heidelberg: Springer, 2006, pp. 265–284.

[5] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, "What can we learn privately?" in *FOCS*. IEEE, Oct 25–28 2008, pp. 531–540.

[6] A. Beimel, K. Nissim, and E. Omri, "Distributed private data analysis: On simultaneously solving how and what," *CoRR*, vol. abs/1103.2626, 2011.

[7] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[8] A. Ambainis, M. Jakobsson, and H. Lipmaa, "Cryptographic randomized response techniques," in *Public Key Cryptography - PKC 2004, 7th International Workshop on Theory and Practice in Public Key Cryptography, Singapore, March 1-4, 2004*, 2004, pp. 425–438.

[9] T. Moran and M. Naor, "Polling with physical envelopes: A rigorous analysis of a human-centric protocol," in *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings*, 2006, pp. 88–108.

[10] X. Cao, J. Jia, and N. Z. Gong, "Data poisoning attacks to local differential privacy protocols," *arXiv preprint arXiv:1911.02046*, 2019.

[11] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings*, 2006, pp. 486–503.

[12] A. Bittau, Ú. Erlingsson, P. Maniatis, I. Mironov, A. Raghunathan, D. Lie, M. Rudominer, U. Kode, J. Tinnes, and B. Seefeld, "Prochlo: Strong privacy for analytics in the crowd," in *Proceedings of the 26th Symposium on Operating Systems Principles*, ser. SOSP '17. ACM, 2017, pp. 441–459.

[13] A. Cheu, A. Smith, J. Ullman, D. Zeber, and M. Zhilyaev, "Distributed differential privacy via shuffling," in *Proceedings of the 38th Annual Conference on the Theory and Applications of Cryptographic Techniques*, ser. EUROCRYPT '19, 2019.

[14] J. Duchi, M. Jordan, and M. Wainwright, "Local privacy and minimax bounds: Sharp rates for probability estimation," in *Advances in Neural and Information Processing Systems 27*, ser. NIPS '13, 2013.

[15] J. Acharya, C. L. Canonne, C. Freitag, and H. Tyagi, "Test without trust: Optimal locally private distribution testing," in *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, ser. AISTATS '19. JMLR, Inc., 2019, pp. 2067–2076.

[16] R. Bassily and A. D. Smith, "Local, private, efficient protocols for succinct histograms," in *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, 2015, pp. 127–135.

[17] J. Duchi, M. Jordan, and M. Wainwright, "Local privacy and statistical minimax rates," in *IEEE 57th Annual Symposium on Foundations of Computer Science*, ser. FOCS '13, 2013, pp. 429–438.

[18] J. Hsu, S. Khanna, and A. Roth, "Distributed private heavy hitters," in *International Colloquium on Automata, Languages, and Programming*. Springer, 2012, pp. 461–472.

[19] R. Bassily, K. Nissim, U. Stemmer, and A. G. Thakurta, "Practical locally private heavy hitters," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, 2017, pp. 2285–2293.

[20] M. Bun, J. Nelson, and U. Stemmer, "Heavy hitters and the structure of local privacy," in *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, Houston, TX, USA, June 10-15, 2018*, 2018, pp. 435–447.

[21] M. J. Kearns, "Efficient noise-tolerant learning from statistical queries," in *STOC*. ACM, May 16-18 1993, pp. 392–401.

[22] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical privacy: the SuLQ framework," in *Proceedings of the 24th ACM Symposium on Principles of Database Systems*, ser. PODS '05. ACM, 2005, pp. 128–138.

[23] R. Bassily, A. Smith, and A. Thakurta, "Private empirical risk minimization: Efficient algorithms and tight error bounds," in *Proceedings of the 55th IEEE Annual Symposium on Foundations of Computer Science*, ser. FOCS '14. Philadelphia, PA: IEEE, 2014, pp. 464–473.

[24] P. Kairouz, S. Oh, and P. Viswanath, "The composition theorem for differential privacy," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul

2015, pp. 1376–1385.

[25] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, B. M. Thuraisingham, D. Evans, T. Malkin, and D. Xu, Eds. ACM, 2017, pp. 1175–1191. [Online]. Available: https://doi.org/10.1145/3133956.3133982

[26] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, R. G. L. D'Oliveira, S. E. Rouayheb, D. Evans, J. Gardner, Z. Garrett, A. Gascón, B. Ghazi, P. B. Gibbons, M. Gruteser, Z. Harchaoui, C. He, L. He, Z. Huo, B. Hutchinson, J. Hsu, M. Jaggi, T. Javidi, G. Joshi, M. Khodak, J. Konecný, A. Korolova, F. Koushanfar, S. Koyejo, T. Lepoint, Y. Liu, P. Mittal, M. Mohri, R. Nock, A. Özgür, R. Pagh, M. Raykova, H. Qi, D. Ramage, R. Raskar, D. Song, W. Song, S. U. Stich, Z. Sun, A. T. Suresh, F. Tramèr, P. Vepakomma, J. Wang, L. Xiong, Z. Xu, Q. Yang, F. X. Yu, H. Yu, and S. Zhao, "Advances and open problems in federated learning," *CoRR*, vol. abs/1912.04977, 2019. [Online]. Available: http://arxiv.org/abs/1912.04977

[27] Ú. Erlingsson, V. Feldman, I. Mironov, A. Raghunathan, K. Talwar, and A. Thakurta, "Amplification by shuffling: From local to central differential privacy via anonymity," in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, T. M. Chan, Ed. SIAM, 2019, pp. 2468–2479. [Online]. Available: https://doi.org/10.1137/1.9781611975482.151

[28] T. Tao, *Topics in Random Matrix Theory*. American Mathematical Society, 2012.

[29] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 13–30, 1963.

Appendix A
Proofs for Section IV

For any integer $d > 2$ and algorithm $R : [d] \to \mathcal{Y}$, let $R(\mathbf{U})$ denote the distribution over $\mathcal{Y}$ induced by sampling $\hat{x}$ from the uniform distribution over $[d]$ and then sampling a message from $R(\hat{x})$. For any set $H \subset [d]$, let $R(\mathbf{U}_H)$ denote the distribution over $\mathcal{Y}$ induced by sampling $\hat{x}$ from the uniform distribution over H and then executing $R(\hat{x})$. In this notation, $Q_{H,R}$ is the algorithm which samples from $R(\mathbf{U}_H)$ when given $+1$, but samples from $R(\mathbf{U}_{\overline{H}})$ when given -1.

In this section, we provide two bounds on the privacy parameter ε' of $Q_{H,R}$ when H is uniformly chosen. The first bound is $O(\varepsilon\sqrt{(\log(|\mathcal{Y}| \cdot |\vec{R}|_{\neq}))/d})$ (Lemma A.5), where $|\mathcal{Y}|$ is the size of the message universe and $|\vec{R}|_{\neq}$ is the number of unique randomizers. The second is $O\big(\varepsilon\sqrt{(\log n)/d}\big)$ (Lemma A.8). We note that the second bound has no dependence on the specification of the randomizers, which may make it looser than the first bound.

The key to the analysis is to argue that, for most messages y and a uniformly random H, the log-odds ratio $\ln(\mathbb{P}[R(\mathbf{U}_H) = y]/\mathbb{P}[R(\mathbf{U}) = y])$ is roughly $\varepsilon/\sqrt{d}$. To this end, we introduce the following definition:

Definition A.1 (Leaky Messages)**.** For any $H \subset [d]$ with size $d/2$ and any local randomizer $R : [d] \to \mathcal{Y}$, a message $y \in \mathcal{Y}$ is *v-leaky with respect to H, R* when

$$\left|\ln \frac{\mathbb{P}[R(\mathbf{U}_H) = y]}{\mathbb{P}[R(\mathbf{U}) = y]}\right| > v \tag{5}$$

Next we show that when y is some fixed message and H is uniformly random, y is $\approx (\varepsilon/\sqrt{d})$-leaky with respect to H, R with low probability.

Claim A.2. *Fix any $\varepsilon > 0$, any $\beta \in (0, 1)$, any $d > 4(e^\varepsilon - 1)^2 \ln \frac{2}{\beta}$, any ε-private $R : [d] \to \mathcal{Y}$. For any message $y \in \mathcal{Y}$, if H is chosen uniformly from subsets of $[d]$ with size $d/2$, then*

$$\mathbb{P}\left[y \textit{ is not } (e^\varepsilon - 1)\sqrt{\frac{4}{d}\ln\frac{2}{\beta}}\textit{-leaky w.r.t. } H, R\right] \geq 1 - \beta$$

Proof. By the definition of leaky message, we must show that the following must hold with probability $\geq 1 - \beta$ over the randomness of H.

$$\left|\ln \frac{\mathbb{P}[R(\mathbf{U}_H) = y]}{\mathbb{P}[R(\mathbf{U}) = y]}\right| \leq (e^\varepsilon - 1)\sqrt{\frac{4}{d}\ln\frac{2}{\beta}} \tag{6}$$

Observe that

$$\begin{aligned}\mathbb{P}[R(\mathbf{U}) = y] &= \sum_{j=1}^{d} \mathbb{P}[R(j) = y] \cdot \mathop{\mathbb{P}}_{x \sim \mathbf{U}}[x = j] \\ &= \sum_{j=1}^{d} \mathbb{P}[R(j) = y] \cdot \frac{1}{d} \qquad \text{(Defn. of } \mathbf{U})\end{aligned}$$

Also observe that, for any fixed choice of H,

$$\mathbb{P}[R(\mathbf{U}_H) = y] = \sum_{i=1}^{d/2} \mathbb{P}[R(h_i) = y] \cdot \frac{2}{d} \quad \text{(By construction)}$$

Now we may write

$$\frac{\mathbb{P}[R(\mathbf{U}_H) = y]}{\mathbb{P}[R(\mathbf{U}) = y]} = \frac{\frac{2}{d}\sum_{i=1}^{d/2} \mathbb{P}[R(h_i) = y]}{\frac{1}{d}\sum_{j=1}^{d} \mathbb{P}[R(j) = y]} \tag{7}$$

For a uniformly random H, observe that each term in the numerator of (7) is a random variable that lies in the interval $(\min_j \mathbb{P}[R(j) = y], e^\varepsilon \min_j \mathbb{P}[R(j) = y])$, due to the ε-privacy guarantee of R. We use the following version of Hoeffding's inequality for samples without replacement.

Lemma A.3 ([29])**.** *Given a set $\vec{p} = \{p_1, \ldots, p_N\} \in \mathbb{R}^N$ such that $p_i \in (c, c')$, if the subset $\vec{x} = \{x_1, \ldots, x_n\}$ is constructed by uniformly sampling without replacement from $\vec{p}$, then*

$$\mathbb{P}\left[\frac{1}{n}\sum_{i=1}^{n} x_i \leq \frac{1}{N}\sum_{i=1}^{N} p_i + (c' - c) \cdot \sqrt{\frac{1}{2n}\log\frac{1}{\beta}}\right] \geq 1 - \beta$$

Hence, the following is true with probability $1-\beta/2$:

$$\begin{aligned}
&(7)\\
&\leq \frac{\frac{1}{d}\sum_{j=1}^{d}\mathbb{P}[R(j)=y]+(e^{\varepsilon}-1)\min_j \mathbb{P}[R(j)=y]\sqrt{\frac{1}{d}\ln\frac{2}{\beta}}}{\frac{1}{d}\sum_{j=1}^{d}\mathbb{P}[R(j)=y]}\\
&= 1+(e^{\varepsilon}-1)\sqrt{\ln\frac{2}{\beta}}\cdot\frac{\sqrt{d}\cdot\min_j \mathbb{P}[R(j)=y]}{\sum_{j=1}^{d}\mathbb{P}[R(j)=y]}\\
&\leq 1+(e^{\varepsilon}-1)\sqrt{\ln\frac{2}{\beta}}\cdot\frac{\sqrt{d}\cdot\min_j \mathbb{P}[R(j)=y]}{d\cdot\min_j \mathbb{P}[R(j)=y]}\\
&= 1+(e^{\varepsilon}-1)\sqrt{\frac{1}{d}\ln\frac{2}{\beta}}\\
&\leq \exp\left((e^{\varepsilon}-1)\sqrt{\frac{1}{d}\ln\frac{2}{\beta}}\right) \qquad (8)
\end{aligned}$$

By a completely symmetric argument, the following holds with probability $1-\beta/2$:

$$\begin{aligned}
(7) &\geq 1-(e^{\varepsilon}-1)\sqrt{\frac{1}{d}\ln\frac{2}{\beta}}\\
&\geq \exp\left(-(e^{\varepsilon}-1)\sqrt{\frac{4}{d}\ln\frac{2}{\beta}}\right) \qquad (9)
\end{aligned}$$

(9) follows from the condition that $d > 4(e^{\varepsilon}-1)^2\ln\frac{2}{\beta}$. (6) follows from (8) and (9) (through a union bound). This concludes the proof. □

Now we apply Claim A.2 to the privacy of Q_{H,R_i}.

A. A protocol-dependent bound on ε'

Our first bound on the privacy parameters will be dependent on the structure of the initial randomizers $R_1,\ldots,R_n$ from which the new randomizers $Q_{H,R_1},\ldots,Q_{H,R_n}$ are derived. We use $|\mathcal{Y}|$ to denote the size of the message universe and $|\vec{R}|_{\neq}$ to denote the number of unique randomizers.

The following is immediate from Claim A.2 by applying a union bound over all the unique randomizers in $\vec{R}$ and the message universe $\mathcal{Y}$:

Corollary A.4. *Fix any vector of ε-private randomizers $\vec{R}=(R_1,\ldots,R_n)$ (where every randomizer has the form $R_i:[d]\to\mathcal{Y}$) and any $d > 4(e^{\varepsilon}-1)^2\ln(12|\mathcal{Y}|\cdot|\vec{R}|_{\neq})$. Sample H uniformly at random over subsets of $[d]$ with size $d/2$. The following is true with probability $\geq 5/6$ over the randomness of H: $\forall y\in\mathcal{Y}\ \forall i\in[n]$ y is not $\left((e^{\varepsilon}-1)\sqrt{\frac{4}{d}\ln 12|\mathcal{Y}|\cdot|\vec{R}|_{\neq}}\right)$-leaky w.r.t. H,R_i*

Lemma A.5. *Fix any ε-locally private protocol $\Pi=(\vec{R},A)$ (where every randomizer has the form $R_i:[d]\to\mathcal{Y}$) and $d > 4(e^{\varepsilon}-1)^2\ln(12|\mathcal{Y}|\cdot|\vec{R}|_{\neq})$. Sample H uniformly at random over subsets of $[d]$ with size $d/2$. The following is true with probability $\geq 2/3$ over the randomness of H: all randomizers $\{Q_{H,R_i}\}_{i\in[n]}$ specified by Algorithm 2 satisfy ε'-privacy, where*

$$\varepsilon' = (e^{\varepsilon}-1)\sqrt{\frac{16}{d}\ln\left(12|\mathcal{Y}|\cdot|\vec{R}|_{\neq}\right)}$$

Proof. From Corollary A.4, all possible outputs of all randomizers are not leaky with probability $\geq 5/6$. More formally, for every $y\in\mathcal{Y}$ and $i\in[n]$,

$$\left|\ln\frac{\mathbb{P}[R_i(\mathbf{U}_H)=y]}{\mathbb{P}[R_i(\mathbf{U})=y]}\right| < \varepsilon'/2$$

By identical reasoning, with probability $\geq 5/6$,

$$\left|\ln\frac{\mathbb{P}[R_i(\mathbf{U}_{\overline{H}})=y]}{\mathbb{P}[R_i(\mathbf{U})=y]}\right| < \varepsilon'/2$$

Recall the definition of Q_{H,R_i}: on input $+1$, it samples from $R_i(\mathbf{U}_H)$ and, on input -1, it samples from $R_i(\mathbf{U}_{\overline{H}})$. From a union bound, we can conclude that the log-odds ratio is at most ε' with probability $\geq 2/3$. This concludes the proof. □

B. A protocol-independent bound on ε'

In this subsection, we obtain a bound on the amplified privacy that depends on the number of users in the protocol but not on the specification of the randomizers $\vec{R}$. If H is drawn uniformly and d is sufficiently large, then for most users, the probability that $R_i(\mathbf{U})$ is a leaky message is small. Let $\mathit{Leak}(v,H,R)=\{y\in\mathcal{Y}\mid y \text{ is } v\text{-leaky with respect to } H,R\}$

Claim A.6. *Fix any $\varepsilon>0$, any $\beta\in(0,1)$, any $d > 4(e^{\varepsilon}-1)^2\ln\frac{2}{\beta}$, and any n algorithms $R_1\ldots R_n$ that are ε-private. If H is sampled uniformly from subsets of $[d]$ with size $d/2$, then the following holds with probability $\geq 5/6$ over the randomness of H: for all $i\in[n]$,*

$$\mathbb{P}\left[R_i(\mathbf{U})\in \mathit{Leak}\left((e^{\varepsilon}-1)\sqrt{\frac{4}{d}\ln\frac{2}{\beta}},H,R_i\right)\right] < 6\beta n$$

Proof. To prove the claim, we show that for every $i\in[n]$, with probability at least $1-1/6n$ over the randomness of H,

$$\mathbb{P}\left[R_i(\mathbf{U})\in \mathit{Leak}\left((e^{\varepsilon}-1)\sqrt{\frac{4}{d}\ln\frac{2}{\beta}},H,R_i\right)\right] < 6\beta n \qquad (10)$$

Below, we use $\binom{[d]}{d/2}$ as shorthand for the subsets of $[d]$ with size $d/2$. We bound the expectation of the random variable:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{H}[\mathbb{P}[R_i(\mathbf{U})\in \mathit{Leak}(\ldots,H,R_i)]]\\
&= \sum_{H\in\binom{[d]}{d/2}}\binom{d}{d/2}^{-1}\cdot\mathbb{P}[R_i(\mathbf{U})\in \mathit{Leak}(\ldots,H,R_i)]\\
&= \sum_{H\in\binom{[d]}{d/2}}\binom{d}{d/2}^{-1}\cdot\sum_{y\in\mathcal{Y}}\mathbb{1}_{\{y\in \mathit{Leak}(\ldots,H,R_i)\}}\\
&\quad\cdot\mathbb{P}[R_i(\mathbf{U})=y]\\
&= \sum_{y\in\mathcal{Y}}\sum_{H\in\binom{[d]}{d/2}}\binom{d}{d/2}^{-1}\cdot\mathbb{1}_{\{y\in \mathit{Leak}(\ldots,H,R_i)\}}\\
&\quad\cdot\mathbb{P}[R_i(\mathbf{U})=y]\\
&\leq \sum_{y\in\mathcal{Y}}\beta\cdot\mathbb{P}[R_i(\mathbf{U})=y] \qquad \text{(Claim A.2)}\\
&= \beta
\end{aligned}$$

Markov's inequality implies that (10) holds with probability $\geq 1 - 1/6n$. □

Claim A.6 is a bound on the probability that $R_i(\mathbf{U})$ is leaky. Because $\vec{R}$ satisfies differential privacy, it implies a bound on the probability that $R_i(\mathbf{U}_H)$ is leaky.

Corollary A.7. *Fix any $\varepsilon > 0$, any $\beta \in (0,1)$, any $d > 4(e^\varepsilon - 1)^2 \ln \frac{2}{\beta}$, and any n algorithms $R_1 \dots R_n$ that are ε-private. If H is sampled uniformly from subsets of $[d]$ with size $d/2$, then the following holds with probability $\geq 5/6$ over the randomness of H: for all $i \in [n]$,*

$$\mathbb{P}\left[R_i(\mathbf{U}) \in \mathit{Leak}\left((e^\varepsilon - 1)\sqrt{\frac{4}{d}\ln\frac{2}{\beta}}, H, R_i\right)\right] < 6\beta n$$
$$\mathbb{P}\left[R_i(\mathbf{U}_H) \in \mathit{Leak}\left((e^\varepsilon - 1)\sqrt{\frac{4}{d}\ln\frac{2}{\beta}}, H, R_i\right)\right] < 6e^\varepsilon \beta n$$

The algorithm Q_{H,R_i} reports either a sample from $R_i(\mathbf{U}_H)$ or from $R_i(\mathbf{U}_{\overline{H}})$. Having bounded the probability that either sample is leaky, we can now argue that Q_{H,R_i} satisfies approximate differential privacy.

Lemma A.8. *Fix any $\varepsilon > 0$, any $\delta, \beta \in (0,1)$, any $d > 4(e^\varepsilon - 1)^2 \ln(12e^\varepsilon n/\delta)$, and any n algorithms that are ε-private. If H is sampled uniformly from subsets of $[d]$ with size $d/2$, then the following holds with probability $> 2/3$ over the randomness of H: all randomizers $\{Q_{H,R_i}\}_{i \in [n]}$ specified by Algorithm 2 satisfy (ε', δ)-privacy, where $\varepsilon' = (e^\varepsilon - 1)\sqrt{\frac{16}{d}\ln(24e^\varepsilon n/\delta)}$.*

Proof. Define $\beta = \delta/(12e^\varepsilon n)$ so that $\varepsilon' = (e^{2\varepsilon} - 1)\sqrt{\frac{16}{d}\ln(2/\beta)}$. For every $Y \subseteq \mathcal{Y}$, the following holds with probability $> 5/6$ by Corollary A.7.

$$\begin{aligned}
&\mathbb{P}[R_i(\mathbf{U}_H) \in Y] \\
&= \mathbb{P}[R_i(\mathbf{U}_H) \in Y - \mathit{Leak}(\varepsilon'/2, H, R_i)] \\
&\quad + \mathbb{P}[R_i(\mathbf{U}_H) \in Y \cap \mathit{Leak}(\varepsilon'/2, H, R_i)] \\
&\leq \mathbb{P}[R_i(\mathbf{U}_H) \in Y - \mathit{Leak}(\varepsilon'/2, H, R_i)] + 6\beta e^\varepsilon n \\
&\hfill \text{(Corollary A.7)} \\
&= \mathbb{P}[R_i(\mathbf{U}_H) \in Y - \mathit{Leak}(\varepsilon'/2, H, R_i)] + \delta/2 \\
&\hfill \text{(Value of } \beta) \\
&= \sum_{y \in Y - \mathit{Leak}(\varepsilon'/2)} \mathbb{P}[R_i(\mathbf{U}_H) = y] + \delta/2 \\
&\leq \sum_{y \in Y - \mathit{Leak}(\varepsilon'/2)} \exp(\varepsilon'/2) \cdot \mathbb{P}[R_i(\mathbf{U}) = y] + \delta/2 \\
&\hfill \text{(Defn. A.1)} \\
&\leq \exp(\varepsilon'/2) \cdot \mathbb{P}[R_i(\mathbf{U}) \in Y] + \delta/2
\end{aligned}$$

By symmetric steps,

$$\mathbb{P}[R_i(\mathbf{U}) \in Y] \leq \exp(\varepsilon'/2) \cdot \mathbb{P}[R_i(\mathbf{U}_H) \in Y] + \delta/2$$

We take identical steps to show that the following holds with probability $> 5/6$ as well:

$$\mathbb{P}[R_i(\mathbf{U}_{\overline{H}}) \in Y] \leq \exp(\varepsilon'/2) \cdot \mathbb{P}[R_i(\mathbf{U}) \in Y] + \delta/2$$
$$\mathbb{P}[R_i(\mathbf{U}) \in Y] \leq \exp(\varepsilon'/2) \cdot \mathbb{P}[R_i(\mathbf{U}_{\overline{H}}) \in Y] + \delta/2$$

From basic composition and a union bound, the following holds with probability $> 2/3$:

$$\mathbb{P}[R_i(\mathbf{U}_{\overline{H}}) \in Y] \leq \exp(\varepsilon') \cdot \mathbb{P}[R_i(\mathbf{U}_H) \in Y] + \delta$$
$$\mathbb{P}[R_i(\mathbf{U}_H) \in Y] \leq \exp(\varepsilon') \cdot \mathbb{P}[R_i(\mathbf{U}_{\overline{H}}) \in Y] + \delta$$

Recall that Q_{H,R_i} samples from $R_i(\mathbf{U}_H)$ on input $+1$ and from $R_i(\mathbf{U}_{\overline{H}})$ on input -1. Hence, Q_{H,R_i} satisfies ε', δ privacy. This concludes the proof. □

Appendix B
A Heavy Hitters Protocol

In this problem, each user has data $x_i \in [d]$. The objective is to find a small subset L of the universe that contains every element $j \in [d]$ such that $\mathit{freq}_j(\vec{x}) > \alpha$. Because there are $1/\alpha$ heavy hitters, the size of L should be $O(1/\alpha)$.

We consider the protocol HH described in [19].

1) Sample public hash function $h : [d] \to [k]$ uniformly from a universal family ($k \ll d$ is a protocol parameter). Also sample π uniformly from partitions of $[n]$ into groups of size $n/\log_2 d$. Intuitively, users in group g will communicate the g-th bit of their data value to the aggregator
2) Each user i in each group g:
   a) obtains $\mathit{bit}(g, x_i)$, the g-th bit in the binary representation of x_i.
   b) computes $x'_i \leftarrow 2 \cdot h(x_i) - \mathit{bit}(g, x_i)$.
   c) reports $y_i \leftarrow R^{\mathtt{HST}}(x'_i)$ to the aggregator.
3) The aggregator iterates through each $j' \in [k]$ and constructs $L_{j'}$ in the following manner:
   a) Iterate through $g \in \log_2 d$. At each step (j', g), gather the messages from group g then use $A^{\mathtt{HST}}$ to obtain an approximate histogram over $2k$. If the estimated frequency of $2 \cdot j' - 1$ is larger than that of $2 \cdot j'$, then set $z_{j',g} \leftarrow 1$ and otherwise $z_{j',g} \leftarrow 0$.
   b) $L_{j'} \leftarrow$ the number represented in binary by $z_{j',1}, \dots, z_{j',\log_2 d}$
4) The aggregator reports $L \leftarrow (L_1, \dots, L_k)$ as heavy hitters

The size of L is at most k and the time spent by the aggregator to construct L is $O(nk^2 \log d)$ (from $k \log_2 d$ executions of $A^{\mathtt{HST}}$). An upper bound on error under manipulation follows from Theorem V.3, taking care to adjust the number of bins to $2k$ and the number of users to $n/\log_2 d$.

Theorem B.1. *For any $\varepsilon \in (0,1)$, any positive integers $m \leq n$, any $\vec{x} = (x_1, \dots, x_n) \in [d]^n$, and any adversary M, if we execute $L \leftarrow \mathrm{Manip}_{m,n}(\mathtt{HH}_\varepsilon, \vec{x}, M)$ with parameter $k \leftarrow 300n^2$, then with probability $\geq 99/100$, L contains all j such that $\mathit{freq}_j(\vec{x}) > \alpha$ where*

$$\alpha = O\left(\sqrt{\frac{(\log d) \cdot \log(n \log d)}{\varepsilon^2 n}} + \frac{m \log d}{\varepsilon n}\right)$$

Proof Sketch. For any group g, let $\vec{x}^{(g)}$ denote the data of users in group g. We first argue that three undesirable events occur with low probability.

- For some g, $|\mathit{freq}(\vec{x}) - \mathit{freq}(\vec{x}^{(g)})|_\infty \gtrapprox \sqrt{(\log d)/n}$. By Hoeffding's inequality and a union bound over all groups, this happens with probability $\leq 1/300$.
- Two data values that appear in $\vec{x}$ collide. Due to the size of k, this happens with probability $\leq 1/300$.
- For some g, the error of the private histogram is too large. Specifically, there is a value $\alpha_0 \approx \sqrt{\frac{(\log d)\cdot\log(n\log d)}{\varepsilon^2 n}} + \frac{m\log d}{\varepsilon n}$ such that $\left\|\hat{\mu}^{(g)} - \mathit{freq}(\vec{x}^{(g)})\right\|_\infty > \alpha_0$. From Theorem V.3 and a union bound over all groups, this event happens with probability $\leq 1/300$.

The remainder of the proof sketch assumes these events have not occurred.

We fix any $j \in [d]$ and any $g \in [\log_2 d]$. We will argue that if j is a heavy hitter, then the aggregator will reconstruct the g-th bit of j. Let $\pi(g)$ be the ordered set of users in group g and let $\vec{x}\,'$ denote the vector $(x'_i)_{i\in\pi(g)}$.

Suppose $\mathit{freq}(j,\vec{x}) = \tau$. Because there are no collisions between hashes, it must be the case that $\mathit{freq}(2h(j) - \mathit{bit}(g,j), \vec{x}\,') \gtrapprox \tau - \sqrt{(\log d)/n}$ and $\mathit{freq}(2h(j) - 1 + \mathit{bit}(g,j), \vec{x}\,') = 0$. The aggregator estimates these frequencies up to simultaneous error α_0. So, when $\tau > \alpha \approx \sqrt{(\log d)/n} + 2\alpha_0$, the estimate of $\mathit{freq}(2h(j) - \mathit{bit}(g,j), \vec{x}\,')$ exceeds that of $\mathit{freq}(2h(j) - 1 + \mathit{bit}(g,j), \vec{x}\,')$. This means the aggregator will assign $z_{h(j),g} \leftarrow \mathit{bit}(g,j)$. □

We remark that the above sketch and analysis are for the simplest version of HH, in which $k = O(n^2)$. In [19], the authors show that $k = O(1/\alpha) = \tilde{O}(\sqrt{n})$ suffices, achieving a smaller list and faster running time. We provide the details of HH in the full version of the paper.

Appendix C
Proof of Theorem V.8

Proof Sketch. Identically to the proof of Theorem V.1, we partition the error contributed by the honest and corrupt users. Let $S \in \{\pm 1\}^{d\times n}$ be the matrix whose columns are $\vec{s}_1, \dots, \vec{s}_n$, and $S_C \in \{\pm 1\}^{d\times |C|}$ be the submatrix consisting only columns corresponding to users $i \in C$. Below, $\sum_i$ stands for $\sum_{i=1}^n$ Then we can write

$$\begin{aligned}&\left\|\tfrac{c\sqrt{d}}{n}\textstyle\sum_i y_i\vec{s}_i - \tfrac{1}{n}\sum_i x_i\right\|_2\\ &= \underbrace{\left\|\tfrac{c\sqrt{d}}{n} S_C(\vec{y}_C - \underline{\vec{y}}_C)\right\|_2}_{\text{manipulation}} + \underbrace{\left\|\tfrac{c\sqrt{d}}{n}\textstyle\sum_i \underline{y}_i\vec{s}_i - \tfrac{1}{n}\sum_{i=1}^n x_i\right\|_2}_{\text{honest execution}}\end{aligned}$$

A lemma from [17] implies that the error introduced by the honest execution of the protocol is $O(\sqrt{d/\varepsilon^2 n})$ with probability $\geq 299/300$.

To bound the error from the manipulation, we will again use bounds on the singular values of the random matrix S_C. As a shorthand, let $c_\varepsilon = \frac{e^\varepsilon+1}{e^\varepsilon-1}$. Then we have

$$\begin{aligned}&\left\|\tfrac{c\sqrt{d}}{n} S_C(\vec{y}_C - \underline{\vec{y}}_C)\right\|_2\\ &\leq \tfrac{c\sqrt{d}}{n}\max_{C\subseteq[n]}\left\|S_C(\vec{y}_C - \underline{\vec{y}}_C)\right\|_2\\ &\leq \tfrac{2c\sqrt{d}}{n}\max_{\substack{C\subseteq[n]\\ |C|=m}}\ \max_{\vec{y}_C\in\{-c_\varepsilon,c_\varepsilon\}^m}\|S_C\vec{y}_C\|_2\\ &= \tfrac{2c\sqrt{d}}{n}\max_{\substack{C\subseteq[n]\\ |C|=m}}\ \max_{\substack{\vec{y}_C\in\mathbb{R}^m\\ \|\vec{y}\|_2\leq c_\varepsilon\sqrt{m}}}\|S_C\vec{y}_C\|_2\\ &\leq \tfrac{2cc_\varepsilon\sqrt{md}}{n}\max_{\substack{C\subseteq[n]\\ |C|=m}}\ \max_{\substack{\vec{y}_C\in\mathbb{R}^m\\ \|\vec{y}\|_2\leq 1}}\|S_C\vec{y}_C\|_2 \qquad (11)\end{aligned}$$

For any $i \in C$, consider the random variable $\vec{s}_i\,' \sim N(0, I_{d\times d})$. The column vector $\vec{s}_i$ is identically distributed with $\frac{\vec{s}_i\,'}{\|\vec{s}_i\,'\|_2}$. By standard concentration arguments, there is a constant c' such that $\min_i\|\vec{s}_i\,'\|_2^2 \geq d - c'\sqrt{d\ln m}$ with probability $\geq 299/300$. In the case where $d < 4(c')^2\ln m$, we bound the error by $2cc_\varepsilon m\sqrt{d}/n = O(m\sqrt{\log n}/\varepsilon n)$. Otherwise, when $d > 4(c')^2\ln m$, we have $\min_i\|\vec{s}_i\,'\|_2^2 > d/2$. Hence,

$$\begin{aligned}(11) &\leq \frac{2cc_\varepsilon\sqrt{md}}{n}\max_{\substack{C\subseteq[n]\\ |C|=m}}\ \max_{\substack{\vec{y}_C\in\mathbb{R}^m\\ \|\vec{y}\|_2\leq 1}}\ \max_{i\in C}\frac{1}{\|\vec{s}_i\,'\|_2}\|S'_C\vec{y}_C\|_2\\ &\leq \frac{cc_\varepsilon\sqrt{8m}}{n}\max_{\substack{C\subseteq[n]\\ |C|=m}}\ \max_{\substack{\vec{y}_C\in\mathbb{R}^m\\ \|\vec{y}\|_2\leq 1}}\|S'_C\vec{y}_C\|_2\\ &= \frac{cc_\varepsilon\sqrt{8m}}{n}\max_{\substack{C\subseteq[n]\\ |C|=m}}\|S'_C\|_2\end{aligned}$$

We apply Lemma V.6 then choose $k = O(\ln n)$ to bound $\|S'_C\|_2$ by $O(\sqrt{d\ln n} + \sqrt{m\ln n})$ with probability $\geq 299/300$. A union bound completes the proof. □