

Distributed Differentially Private Data Analytics via Secure Sketching

Jakob Burkhardt, Hannah Keller, Claudio Orlandi, Chris Schwiegelshohn
{jakob,hkeller,orlandi,schwiegelshohn}@cs.au.dk
Aarhus University
Denmark

Abstract

We introduce the *linear-transformation model*, a distributed model of differentially private data analysis. Clients have access to a trusted platform capable of applying a public matrix to their inputs. Such computations can be securely distributed across multiple servers using simple and efficient secure multiparty computation techniques.

The linear-transformation model serves as an intermediate model between the highly expressive *central model* and the minimal *local model*. In the central model, clients have access to a trusted platform capable of applying any function to their inputs. However, this expressiveness comes at a cost, as it is often prohibitively expensive to distribute such computations, leading to the central model typically being implemented by a single trusted server. In contrast, the local model assumes no trusted platform, which forces clients to add significant noise to their data. The linear-transformation model avoids the single point of failure for privacy present in the central model, while also mitigating the high noise required in the local model.

We demonstrate that linear transformations are very useful for differential privacy, allowing for the computation of linear sketches of input data. These sketches largely preserve utility for tasks such as private low-rank approximation and private ridge regression, while introducing only minimal error, critically independent of the number of clients.

1 Introduction

Differential Privacy (DP) [28] has become the de-facto standard for ensuring the privacy of individuals whose data is used in data analytics. DP offers strong, provable guarantees, such as composability and resilience to auxiliary information. In the classic *central model* of differential privacy, clients submit their data to a trusted central server, which processes the data and releases a (necessarily) noisy version of the result. Analysts can then use this result to perform queries on the data. Informally, differential privacy guarantees that the analyst cannot confidently determine whether any individual contributed their data, except with some small probability.

The central model requires trust that the server will not reveal any individual’s data and cannot be compromised by external actors. However, universally trusted servers may not exist in practice. In situations where privacy is critical but trusting a single server is not feasible, alternative approaches are available: One such approach is to use secure multiparty computation (MPC) techniques to distribute the central server’s computations across multiple servers. Under the assumption that a subset of these servers are honest, this setup guarantees the same level of privacy as the central model. However, MPC techniques often introduce significant computational

overhead, especially when distributing complex or expressive computations.

Another option is to avoid relying on external servers altogether. In this case, each client must publish their data in a locally differentially private manner, by adding noise directly to their data. While this removes the need for a trusted server, it results in lower utility due to the increased noise. This approach is known as the *local model* of differential privacy [45], an example of which is randomized response [76].

Both the central and local models of differential privacy have strengths and drawbacks. The central model typically yields higher utility but requires strong trust in a single server or the costly use of MPC. The local model, by contrast, requires no trust in external parties but suffers from reduced utility. These tradeoffs have motivated significant research into finding a middle ground between these two models. One promising direction is to limit the expressiveness of the class of functions \mathcal{F} that can be executed in a trusted manner. While this restriction allows for more efficient distributed implementation, it is only useful if the DP mechanism built around such functions can still achieve good accuracy.

A notable example of such restricted expressiveness is the class of functions known as *shuffles*, where a central entity (or a distributed system) randomly permutes the messages from clients before they reach the analysing server. This is known as the *shuffle model* [12, 18, 31]. The shuffle model has been used to implement many differentially-private mechanisms with reasonable utility [18, 31]. Surprisingly, the shuffle model is more expressive than initially expected: with a trusted shuffler, any function can be securely computed in two rounds under the assumption of an honest-majority [10].

We therefore ask the following:

Question 1.1. *What is the least expressive class of functions \mathcal{F} that needs to be securely implemented in order to achieve computationally efficient distributed differential privacy with utility comparable to that of the central model?*

1.1 Our Contribution

In this work, we investigate the power of the *linear-transformation model* (LTM) for differentially-private mechanisms. In this model, the clients only have access to a trusted platform for performing arbitrary linear transformations of their collective inputs. Linear functions can be distributed extremely efficiently with secure computations. However, it is also known that the expressiveness of this class of functions is strictly limited.

For a visualization of the architecture, we refer to Figure 1. We demonstrate the benefits of this linear transformation model (LTM) by showing that it can be used, in combinations with *linear sketches*

to construct private summaries of a data set with noise comparable to that of central differentially private mechanisms. As in the local model, each client adds some amount of noise to their data before passing it onto the linear sketch. We then release the linear transform of the data. The benefit of having a trusted linear sketch is that the noise can be combined, thereby significantly reducing the amount of noise per client. All of our mechanisms require only a single round of communication from clients to server.

We demonstrate the effectiveness of this model by exemplarily applying it to problems in numerical linear algebra. A particularly powerful and expressive class of linear sketches with numerous linear algebra applications [77] are Johnson-Lindenstrauss (JL) transforms. In this paper, we focus on *low rank approximation* and *regularized linear regression*. Our contribution is to show how to design mechanisms for these problems, with the most widely used JL-transforms as the underlying secure sketch. For these problems, we achieve utility comparable to that of the central model with local privacy guarantees, assuming the JL-transform is computed securely.

For both of the following problems, we have the option of achieving ϵ differential privacy, via protocols based on the Laplace mechanism, as well as (ϵ, δ) -differential privacy with slightly higher utility, via protocols based on the Gaussian mechanism.

Low Rank Approximation. We are given a data matrix $\mathbf{A} \in \mathbb{R}^{n \times d}$. Our goal is to compute an orthogonal projection $\mathbf{X} \in \mathbb{R}^{d \times r}$ minimizing the error $OPT = \min_{\mathbf{X}} \|\mathbf{A} - \mathbf{A}\mathbf{X}\mathbf{X}^T\|_F$, where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. In the LTM, we give a (ϵ, δ) differentially private mechanism with multiplicative error $(1 + \alpha)$ and additive error $\tilde{O}(kd^{3/2}\alpha^{-3}\epsilon^{-1}\log\delta^{-1})$. We also present a ϵ differentially private mechanism with multiplicative error $(1 + \alpha)$ and additive error $\tilde{O}(k^3d^{3/2}\alpha^{-6}\epsilon^{-1})$. State of the art central model mechanisms [30, 39] have no multiplicative error and additive error $O(k\sqrt{d} \cdot \epsilon^{-1})$, which is optimal in terms of k and d . For the local model, we are only aware of the result by [3], which has a multiplicative error of $O(\log n \log^3 d)$ and additive error $O(k^2 n \cdot d \text{poly}(\epsilon^{-1}, \log \delta^{-1}))$. There are several related questions we will survey in more detail in Section 2.

Ridge Regression. We are given a data matrix $\mathbf{A} \in \mathbb{R}^{n \times d}$, a target vector $\mathbf{b} \in \mathbb{R}^n$ and regularization parameter $\lambda > 0$. Our goal is to find $\mathbf{x} \in \mathbb{R}^d$ minimizing $\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda \cdot \|\mathbf{x}\|^2$, where $\|\cdot\|$ denotes the Euclidean norm of a vector. Each row of the data matrix, as well as the corresponding entry of \mathbf{b} resides with a client. Assume that $\lambda \geq \text{poly}(\epsilon^{-1}, d, \log 1/\delta)$. In the LTM, there exists an (ϵ, δ) differentially private mechanism that can compute an $\hat{\mathbf{x}}$ with error $(1+o(1)) \cdot (\|\mathbf{A}\mathbf{x}_{opt} - \mathbf{b}\|^2 + \lambda \cdot \|\mathbf{x}_{opt}\|^2) + \text{poly}(\epsilon^{-1}, d, \log 1/\delta)$. Unlike the low rank approximation problem, we are not aware of other papers achieving comparable bounds in any model. Regression has been studied in the context of empirical risk minimization and some results also exist for comparing closeness of the computed predictor $\hat{\mathbf{x}}$ and that optimal \mathbf{x}_{opt} . While we view the low rank approximation mechanisms as our main result, we nevertheless believe this work can serve as a starting point for achieving bounds for ridge regression in terms of cost in any model including the central model.

Table 1 provides a comparison of the utility bounds for the problems in the local, central, shuffle and LTM models. To the best of

our knowledge, there are no results on low rank approximation in the shuffle model.

As a warm-up to illustrate the potential inherent in the LTM, and to also highlight the comparison to the other models, we consider the basic problem of computing histograms and frequency of moments.

2 Related Work

There is substantial work on the shuffle model, which also aims at facilitating differential privacy in a distributed setting. We discuss the relationship between the LTM and the shuffle model in more detail in Section 4. For the problems we studied here, there is an abundance of prior work discussed as follows.

Low Rank Approximation: There are various ways in which one could formulate the low rank approximation problem. The setting which is most important to us is the seminal paper [30], who achieved a worst case additive error of the order $k\sqrt{d}$ for outputting an orthogonal projection matrix V_k in the row space of \mathbf{A} . This bound is also optimal. In the local model where each client holds a row of the data matrix \mathbf{A} , and we wish to output an orthogonal projection in the column space of \mathbf{A} , [68] gave an algorithm with an additive error of the order \sqrt{n} , which matches the lower bound by [7] up to lower order terms. This is different from computing a projections in the row space of \mathbf{A} considered by [30] and indeed our own results. The only known local algorithm to do likewise in the local model is due to [3], with a substantially larger multiplicative and additive error compared to our results. To the best of our knowledge, no single round shuffle protocol improving over the local bounds is known.

Low rank approximation and its sister problem PCA, have seen substantial attention in settings more loosely related to our work. Much of the work on private low rank approximation [5, 14, 17, 30, 38–40, 44] considers the data to be fixed, often using Gaussian noise. In addition, a strong spectral gap assumption, typically of the form that the first singular value is substantially larger than the second, is crucial to the analysis. We note that all algorithms operating with this assumption do not yield worst case bounds.

When there is no spectral gap, we compare to the central model exponential mechanism approach from [17], implemented by [44]. [52] also provide a solution to this problem; however, it is unfortunately computationally intractable.

Ridge Regression. The previous work most related to ours is due to [63]. The author gave a mechanism that preserved the entire spectrum of a data matrix in a private manner and also showed that the returned regression vector $\hat{\mathbf{x}}$ is close to the optimal \mathbf{x} assuming that the matrix is well conditioned.

Most previous work [8, 16, 49, 52, 55, 69, 71–75, 79] study private linear regression in the context of risk minimization, where the algorithm is given i.i.d. samples from some unknown distribution and aims at computing a solution with good out of sample performance. Even without privacy constraints risk minimization yields an additive error that depends on n . Another line of work studies various loss functions, including regression in a Bayesian setting [26, 33, 56]. We are not aware of any prior work on linear regression in the local DP model.

Table 1: SOTA utility bounds for various models for frequency estimation and low rank approximation. Our bounds for low rank approximation have a multiplicative error $O(1 + \alpha_S)$, which we omitted for space.

	Local	Shuffle	LTM (this work)	Central
Frequency Est	$\tilde{O}(\sqrt{n})$ [6] $\Omega(\sqrt{n})$ [7]	$\tilde{\Theta}(\varepsilon^{-1})$ [34]	$\tilde{O}(\varepsilon^{-1})$	$\tilde{\Theta}(\varepsilon^{-1})$
Low-Rank App	$(\tilde{O}(\log n \log^3 d), O(k^2 nd))$ [3] $\Omega(\sqrt{n})$ [7]	N.A.	$\tilde{O}(\frac{k d^{3/2}}{\alpha_S^3} \varepsilon^{-1})$	$\tilde{\Theta}(k \sqrt{d} \varepsilon^{-1})$ [30]

DP via Johnson-Lindenstrauss Transforms. The Johnson-Lindenstrauss lemma has been previously used in differential privacy. [13] studies the DP guarantee a JL transform itself gives in the central model, in the context of cut queries and directional variance queries. [48] and [64] use JL transforms together with Gaussian noise in the context of DP Euclidean distance approximation, to decrease the error. This was subsequently improved in [65], using Laplace noise. The difference to our work is that the transform is used to reduce d instead of n , by having every client apply it locally before adding noise. In [59] the authors make use of JL transforms to achieve private query release, by applying it before releasing a bundle of queries. [35] studies pairwise statistics in the local model of differential privacy and uses JL transforms to reduce client-sided dimensionality.

Differentially Private Non-Oblivious Sketches. Non-Oblivious sketches, such as CountSketch, have also been used in combination with either distributed or central DP. [60, 78] provide an analysis of differentially private CountSketch in the central model. [53] consider differentially private CountSketch and Count-Min Sketch in a distributed setting, but they reduce the dimensionality of each input, which can be done by users locally, rather than reducing the dependence on the number of users, as we do.

3 Preliminaries

Notation. Column vectors are written in bold lowercase letters \mathbf{b} and matrices in bold uppercase letters \mathbf{A} . The transpose operator over vectors and matrices is \mathbf{b}^T and \mathbf{A}^T . For any vector \mathbf{b} , we denote by $\|\mathbf{b}\| = \sqrt{\sum_i b_i^2}$ its ℓ_2 -norm. For any matrix \mathbf{A} , we denote by $\|\mathbf{A}\|_F = \sqrt{\sum_i \sum_j A_{i,j}^2}$ its Frobenius norm. We denote the inner product between vectors $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^T \mathbf{b}$. Two vectors \mathbf{a}, \mathbf{b} are orthogonal if their inner product is 0. An orthogonal matrix \mathbf{A} is a real square matrix where the columns have unit Euclidean norm and are pairwise orthogonal.

Subspace Preserving Sketches. We consider subspace-preserving sketches in this work, which are sparse Johnson-Lindenstrauss transforms. Of particular interest for us are algorithms that compute a subspace approximation of A without prior knowledge of A . Such algorithms are known as oblivious subspace embeddings. We will use a family of such embeddings known as *oblivious sparse norm-approximating projections* originally due to [58], but since improved upon in several subsequent works.

Definition 3.1 (OSNAP [58]). Let $\mathcal{D}_{m,n,s}^{\text{sketch}}$ be a distribution over random matrices $\{-1, 0, 1\}^{m \times n}$, where s entries of each column, chosen uniformly and independently for each column, are set to ± 1 with equal probability, and all other entries are set to 0. If $s = 1$, we

write $\mathcal{D}_{m,n}^{\text{sketch}}$. We say that $S \sim \mathcal{D}_{m,n,s}^{\text{sketch}}$ is an (α, β, s, m) OSNAP for a $n \times d$ orthogonal matrix \mathbf{W} if with probability $1 - \beta$, for all $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$ and some precision parameter α

$$\left| \mathbf{a}^T \mathbf{W}^T \mathbf{S}^T \mathbf{S} \mathbf{W} \mathbf{b} - \mathbf{a}^T \mathbf{W}^T \mathbf{W} \mathbf{b} \right| \leq \alpha \cdot \|\mathbf{W} \mathbf{a}\| \|\mathbf{W} \mathbf{b}\|.$$

We give bounds on available choices of (α, β, s, m) for subspaces of rank k in Table 2.

Oblivious subspace embeddings (OSEs) [62], allow for faster approximation algorithms for problems in linear algebra. Achlioptas [1] provided the first Johnson-Lindenstrauss transform with some amount of sparsity. The first embedding with asymptotically smaller number of non-zeros than dense Johnson-Lindenstrauss transforms was probably due to [43], who also applied them in the context of subspace embeddings. In a remarkable result, the super-sparse version with only 1 non-zero entries per column, was first analyzed in [21] and improved independently in [54, 58], by increasing the target dimension m . In [58], the sparse Johnson-Lindenstrauss transform [43] was studied with a wider range of parameters. Subsequent works [22, 42] further analyze and improve the relationship between m and s , emphasizing different parameters. Table 2 provides the exact interplay between m and s in those works. [51] proposes an approach to boost the success probability β of an OSE, which gives an alternative to increasing the target dimension.

When dealing with rank k approximation, we will condition on

$$(1 - \alpha_S) \|\mathbf{A} - \mathbf{A} \mathbf{X} \mathbf{X}^T\|_F^2 \leq \|\mathbf{S}(\mathbf{A} - \mathbf{A} \mathbf{X} \mathbf{X}^T)\|_F^2 \leq (1 + \alpha_S) \|\mathbf{A} - \mathbf{A} \mathbf{X} \mathbf{X}^T\|_F^2$$

for all rank k orthogonal matrices \mathbf{X} . When dealing with regression, we will condition on

$$(1 - \alpha_S) \|\mathbf{A} \mathbf{x}' - \mathbf{b}\|^2 \leq \|\mathbf{S}(\mathbf{A} \mathbf{x}' - \mathbf{b})\|^2 \leq (1 + \alpha_S) \|\mathbf{A} \mathbf{x}' - \mathbf{b}\|^2$$

for all $\mathbf{x} \in \mathbb{R}^d$. For the parameters given here, this is true with the probability given in Table 2 (assuming $k = d$ in the case of regression). For the case of regression in particular, it is sometimes beneficial to select α_S as large as possible. A sufficiently largest value of α_S such that \mathbf{S} still provide a subspace embedding guarantee is $1/3$. Throughout this paper, we will sometimes bound α_S by 1.

We further will give our utility proofs using the sparsity/target dimension bounds from [22]. Other tradeoffs are possible, but they have worse bounds for most ranges of parameters. The results in [42, 54, 58] give different trade-offs depending on which set of parameters are considered the most dominant. Notably, the $s = 1$ sketches of [54, 58] result in an additive error of $\tilde{O}\left(\frac{k^4 d^3}{\alpha_S^4 \beta^2} \varepsilon^{-2} \log \frac{1}{\delta}\right)$ for low

Table 2: Trade-off between the target dimension m and sparsity s of sparse OSEs that are generated such that they have s non-zeros entries per column. Here α denotes the accuracy of the OSE and β denotes its fail probability. Variable k is the parameter to k -rank approximation, which can be replaced by dimension d for linear regression. The paper by [1] was the first to propose the construction, while its application to preserving subspaces was given by [20].

Paper	Target Dimension m	Sparsity s	With Probability
[1]	$O(\frac{k+\log 1/\beta}{\alpha^2})$	m	$1 - \beta$
[58]	$O(\frac{k^2}{\alpha^2\beta})$	1	$1 - \beta$
[22]	$O(\frac{k \log(k/\beta)}{\alpha^2})$	$O(\frac{\log(k/\beta)}{\alpha})$	$1 - \beta$
[42]	$O(\frac{k}{\alpha^2})$	$O(\frac{1}{\alpha} \cdot (\frac{k}{\log(1/\alpha)} + k^{2/3} \log^{1/3}(k)))$	$1 - 2^{-k^{2/3}}$

rank approximation and a term $\tilde{O}\left(\frac{d^7 \epsilon^{-2} \log \frac{1}{\delta}}{\alpha_s^2 \beta^2 \lambda} + \frac{d^{14} \epsilon^{-4} \log^2 \frac{1}{\delta}}{\alpha_s^6 \beta^4 \lambda^2}\right)$ in both the multiplicative and additive error. The utility guarantees using the bounds from [42] change only by logarithmic factors compared to those in Theorems 5.4 and 5.8.

Differential Privacy. Differential privacy [28] offers privacy guarantees to individuals contributing their data to some randomized algorithm. We say that two datasets are neighboring if one can be obtained from the other by the replacement of a single individual with another individual.

Definition 3.2 (Differential Privacy in the central model [28]). Let $\epsilon \geq 0$ and $\delta \in [0, 1]$. A randomized mechanism $\mathcal{M} : \mathcal{X} \rightarrow \mathcal{Y}$ is (ϵ, δ) -differentially private, if for all neighboring data sets $x, x' \in \mathcal{X}$ and all outputs $S \subseteq \mathcal{Y}$ it holds that

$$\Pr[\mathcal{M}(x) \in S] \leq e^\epsilon \Pr[\mathcal{M}(x') \in S] + \delta,$$

where the probabilities are over the randomness of \mathcal{M} . If $\delta = 0$, then we simply say that the mechanism is ϵ -differentially private.

We first define the Gaussian and Laplace mechanisms, which we use later.

Lemma 3.3 (The Gaussian Mechanism [29]). Let $f : \mathcal{X} \rightarrow \mathbb{R}^k$ be a function and let $\epsilon \geq 0$ and $\delta \in [0, 1]$. The Gaussian mechanism adds to each of the k components of the output, noise sampled from $N(0, \sigma^2)$ with

$$\sigma^2 \geq \frac{2(\Delta_2 f)^2 \ln(1.25/\delta)}{\epsilon^2},$$

where $\Delta_2 f = \max_{x \sim x'} \|f(x) - f(x')\|_2$ denotes the ℓ_2 sensitivity of function f . The Gaussian mechanism is (ϵ, δ) differentially private.

Lemma 3.4 (The Laplace Mechanism [28]). Let $f : \mathcal{X} \rightarrow \mathbb{R}^k$ be a function and let $\epsilon \geq 0$. The Laplace mechanism adds to each of the k components of the output, noise sampled from $\text{Lap}(\Delta_1 f / \epsilon)$, where $\Delta_1 f = \max_{x \sim x'} |f(x) - f(x')|$ denotes the ℓ_1 sensitivity of function f . The Laplace mechanism is ϵ differentially private.

We will use sequential composition of differentially private mechanisms.

Lemma 3.5 (Sequential Composition [28]). Let $\mathcal{M}_i : \mathcal{X} \rightarrow \mathcal{Y}_i$ be an (ϵ_i, δ_i) -differentially private mechanism for $i \in [k]$. Then mechanism $\mathcal{M} : \mathcal{X} \rightarrow \prod_{i=1}^k \mathcal{Y}_i$ defined as $\mathcal{M}(x) = (\mathcal{M}_1(x), \dots, \mathcal{M}_k(x))$, is $(\sum_{i=1}^k \epsilon_i, \sum_{i=1}^k \delta_i)$ -differentially private.

A probability distribution F is *infinitely divisible* if for any positive integer n , there is a random variable S_n which can be written as $S_n = \sum_{i \in [n]} X_i$ such that each of the X_i are random variables, independent and identically distributed, and S_n has the probability distribution F .

4 Privacy Guarantees in the LTM

We consider a model in which the trusted component can perform any public linear transformation of the inputs. The model includes three algorithms: (1) $R : \mathcal{X} \rightarrow \mathcal{Y}$ is a randomized encoder that takes a single user's data and outputs a randomized message, (2) $T : \mathcal{Y}' \rightarrow \mathcal{Y}^*$ is the idealized trusted component that performs a public linear transformation of its inputs, and (3) $A : \mathcal{Y}^* \rightarrow \mathcal{Z}$ is an analysis function that takes the results messages and estimates some function from these messages.

Note that standard definitions of differential privacy in the shuffle model only require $T(R(x_1), \dots, R(x_n))$ to be differentially private. This implicitly assumes that all clients are honest and do not collude with the adversary, in particular they are assumed not to leak the output of their randomizers publicly. This implies that existing definitions of differential privacy in the shuffle model could be satisfied even by (artificial) mechanisms in which a single client adds the whole noise while the others do not randomize their messages at all. This is clearly a very weak privacy guarantee: in a setting in which a large number of clients participate in a differentially private data analysis, it is unrealistic to assume that the adversary does not control even a single client. Luckily, to the best of our knowledge, no proposed mechanisms in the literature suffer from these vulnerabilities, still this counterexample shows that the existing definition is not robust enough, and we therefore formalize a notion of differential privacy where we explicitly tolerate that a bounded number of clients might collude with the adversary. Similar observations were also made by [67].

Definition 4.1 (Trusted Computation Model for Differential Privacy). A tuple of algorithms $P = (R, T, A)$ is (ϵ, δ) -differentially private given corrupt clients C_{cor} if the output $\Pi_R(x_1, \dots, x_n) = T(R(x_1), \dots, R(x_n))$, as well as corrupted parties' randomizer output $R(x_i)$ for all $x_i \in C_{cor}$ satisfy (ϵ, δ) -differential privacy.

Multi-Central Model of Differential Privacy.

[66] introduce the multi-central model, which exactly defines this split-trust model that instantiates the trusted computation model above. They allow a semi-honest adversary that honestly follows

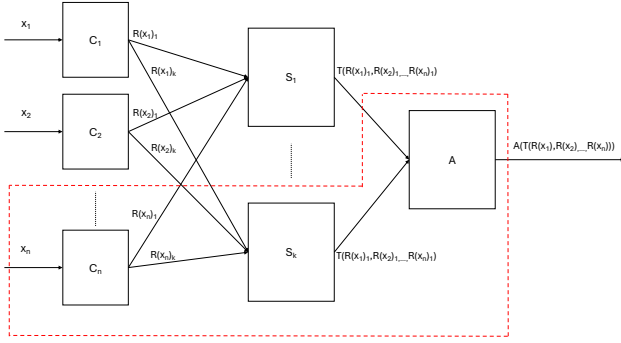


Figure 1: Adversary's View

the protocol to corrupt up to all but one servers and all but one client. We also operate in the same semi-honest setting, but define parameters t and t' for the number of tolerated server and client corruptions.

The view $\text{View}_{C_{cor}, S_{cor}}^{R, \Pi, A}(x)$ of the adversary consists of all the information available to the corrupted clients C_{cor} and servers S_{cor} plus the final output of the honest, uncorrupted servers. This view excludes only internal information of the trustworthy servers and clients. The resulting protocol is (ϵ, δ) -differentially private in the multi-central model if the adversary's view is (ϵ, δ) -indistinguishable from the output on a neighboring dataset.

Figure 1 further illustrates what is contained in the adversary's view. If one server and one client are corrupted, the adversary's view contains all incoming and outgoing messages from that server and that client marked in red in the figure.

Definition 4.2 (Instantiation of Trusted Computation Model for Differential Privacy with MPC). Let Π be a k -party MPC protocol that computes $f : \mathbb{R}^n \rightarrow \mathbb{R}$ with perfect security, tolerating t corruptions. A tuple of algorithms $P = (R, \Pi, A)$ is (ϵ, δ) -differentially private if for all coalitions S_{cor} of up to $t < k$ corrupt servers and for all possible S , all coalitions C_{cor} of $t' < n$ corrupt clients and all neighboring datasets x and x' :

$$\begin{aligned} \Pr[\text{View}_{S_{cor}, C_{cor}}^{R, \Pi, A}(x) \in S] \\ \leq e^\epsilon \cdot \Pr[\text{View}_{S_{cor}, C_{cor}}^{R, \Pi, A}(x') \in S] + \delta \end{aligned}$$

where the probability is over all the randomness in the algorithms (R, Π, A) .

Algorithm 1 describes how clients and servers jointly compute the linear transformation, allowing the transformed data to be published and used for analysis.

With this definition, we can give our first result showing that differential privacy is retained given a bounded number of corruptions.

Lemma 4.3. *If $P = (R, T, A)$ is (ϵ, δ) -differentially private with t' corrupt clients, and if Π is a perfectly secure k -party MPC protocol that computes T correctly while tolerating t corrupt servers, then $P = (R, \Pi, A)$ is (ϵ, δ) -differentially private.*

PROOF SKETCH. This proof can be seen as analogous to the security proof of Theorem 4 in [67]. Recall that each server learns

Algorithm 1 Linear Transformation Model

- 1: **Input:** Individual input vectors x_1, \dots, x_n representing values in \mathbb{R}^d
 - 2: Each client $i \in [n]$ locally computes $R(x_i)$ on their input x_i .
 - 3: Each client secret shares the resulting value using a linear secret sharing scheme, and sends one share to each server.
 - 4: The servers jointly compute function T on the matrix, which results from concatenating all n vector secret shares.
 - 5: The resulting matrix product is published and then taken as input to any analysis function A .
 - 6: **Output:** Output $A(T(R(x_1), \dots, R(x_n)))$
-

nothing more from a perfectly secure multi-party computation protocol other than what is implied by their own input and the output of the function being evaluated. Then there is a simulator $\text{Sim}_{S_{cor}, C_{cor}}^{R, \Pi, A}(x)$ whose output is (ϵ, δ) indistinguishable from the adversary's view $\text{View}_{S_{cor}, C_{cor}}^{R, \Pi, A}(x)$. Note that the secret shares that are part of the view of the corrupted servers in the protocol Π are independent of the input values of the corrupted clients, and all can be easily simulated by random sampling.

A bit more formally, let x be an input dataset, where x_i is the input value contributed by client i and \tilde{x}_i is the vector of input shares for server i that serves as input to Π . There is a mapping from $x \in \mathcal{X}^n$ to $\tilde{x} \in \tilde{\mathcal{X}}^k$. Denote $X = \{x_i\}_{i \in C_{cor}}$ and $X' = \{x'_i\}_{i \in C_{cor}}$. Then for any two datasets x, x' that differ in one entry x_i with $i \notin C_{cor}$:

$$\begin{aligned} \Pr[\text{View}_{S_{cor}, C_{cor}}^{R, \Pi, A}(x) \in S] \\ \leq \Pr[\text{Sim}_{S_{cor}, C_{cor}}^{R, \Pi, A}(f(R(x_1), \dots, R(x_n)), X) \in S] \\ \leq e^\epsilon \Pr[\text{Sim}_{S_{cor}, C_{cor}}^{R, \Pi, A}(f(R(x'_1), \dots, R(x'_n)), X') \in S] + \delta \\ = e^\epsilon \Pr[\text{View}_{S_{cor}, C_{cor}}^{R, \Pi, A}(x') \in S] + \delta \end{aligned}$$

□

4.1 Frequency Moments

To illustrate the possibilities inherent to the LTM, we consider a simple application by way of estimating frequency moments. For simplicity, assume in this section that all clients are trustworthy, i.e. $t' = 0$, but the arguments can be straightforwardly extended to deal with arbitrary values of t' . Here, each client is given a number $[-\Delta, \Delta]$ and our goal is to estimate $F_k = \sum_{i=1}^n |x_i|^k$. In the local model, even if all entries are either 0 or 1, it is not possible to estimate F_1 without incurring an additive error of the order \sqrt{n} , that is the estimated value $\tilde{F}_1 = F_1 \pm O(\sqrt{n})$ for any privacy preserving local mechanism [7]. In contrast, simply computing F_k and adding an appropriate amount of noise yields an additive error that only depends on Δ, ϵ and δ .

We now consider the LTM. Let \mathbf{x}^k denote the vector of client entries with $x_i^k = |x_i^k|$. We observe that $F_k = f_k(\mathbf{x}) = 1^T \mathbf{x}^k$. Suppose that every client i samples g_i from a distribution, and we denote $\tilde{\mathbf{x}}$ the vector of noisy client entries with $\tilde{x}_i^k = |x_i^k| + g_i$. To guarantee differential privacy, we can use a differentially private additive noise mechanism $1^T(\tilde{\mathbf{x}}^k) = f_k(\mathbf{x}) + G$, where the noise distribution is infinitely divisible and $G = \sum_{i \in [n]} g_i$.

Many different infinitely divisible distributions can be used at this stage. One example is the geometric distribution, since the sum of negative binomial random variables follows a geometric distribution, which can be used in differentially private mechanisms [36]. The Laplace distribution, which is commonly used for ϵ -DP mechanisms, is also infinitely divisible, since a zero-centered Laplace distribution can be formulated as a sum of differences between Gamma distributions. Other possible options are stable distributions, a subset of infinitely divisible distributions for which each g_i and G follow the same distribution, such as Gaussian or Cauchy.

Since Gaussians are popular in differentially private mechanisms, we will describe the parameter choices for this distribution. Let g_i be Gaussian distributed with zero mean and variance $\sigma^2 \geq \frac{2\Delta^k \ln(1.25/\delta)}{n \cdot \epsilon^2}$. Then we have $\tilde{F}_k = 1^T(\mathbf{x}^k + g) = F_k + \sum_{i=1}^n g_i$. We now observe that $\sum_{i=1}^n g_i$ is Gaussian distributed with zero mean and variance $n \cdot \sigma^2 \geq \frac{2\Delta^k \ln(1.25/\delta)}{\epsilon^2}$. Thus, assuming a trusted computation of $1^T g$, \tilde{F}_k is a differentially private estimate of F_k with error $\frac{2\Delta^k \ln(1.25/\delta)}{\epsilon^2}$.

We also describe the analogous parameter choices for the Laplace mechanism. Let g_i be sample from the distribution defined as the difference between two Gamma distributions $\Gamma(\frac{1}{n}, \frac{2\Delta^k}{\epsilon})$. Again, we have $\tilde{F}_k = 1^T(\mathbf{x}^k + g) = F_k + \sum_{i=1}^n g_i$. We now observe that $\sum_{i=1}^n g_i$ is Laplace distributed, centered at 0 and with scale $\frac{2\Delta^k}{\epsilon}$. Thus, assuming a trusted computation of $1^T g$, \tilde{F}_k is a differentially private estimate of F_k with error $\frac{2\Delta^k}{\epsilon}$.

4.2 Privacy Preserving Mechanism for Oblivious Johnson Lindenstrauss Transforms

We instantiate the trusted computation model for differential privacy for specific choices for the functions in tuple (R, T, A) . We will show how to guarantee differential privacy for the tuples $(R_D^{\text{dense}}, T_S^{\text{dense}}, A)$ and (R_D, T_S, A) , which we will refer to as the dense sketching mechanism and the sparse sketching mechanism, respectively.

Randomizer function $R_D : \mathbb{R}^d \rightarrow \mathbb{R}^{ds}$. The randomizer takes as input a d -dimensional row vector \mathbf{x} , copies it s times and concatenates the result to form \mathbf{x}^s . Then noise is sampled independently from distribution D and added to each of the ds entries in the resulting vector.

$$R_D^{\text{dense}}(\mathbf{x}) = \mathbf{x}^s + \mathbf{g}, \mathbf{g} \sim D^{ds} \quad (1)$$

We define a second alternative to this randomizer function, which instead outputs 0^{ds} if $n < 8m \ln(dm/\delta) + t'$:

$$R_D(\mathbf{x}) = \begin{cases} 0^{ds} & n < 8m \ln(dm/\delta) + t' \\ R_D^{\text{dense}}(\mathbf{x}) & \text{else} \end{cases} \quad (2)$$

Transformation function $T_S : \mathbb{R}^{n \times ds} \rightarrow \mathbb{R}^{m \times d}$. This function takes as a parameter a matrix $\mathbf{S} \in \{-1, 0, 1\}^{m \times n}$ and takes as input a matrix $\mathbf{A} \in \mathbb{R}^{n \times ds}$, which corresponds to the vertical concatenation or stacking of n outputs from a randomizer function R . We parse \mathbf{A} as the horizontal concatenation of matrices $\mathbf{A}_i \in \mathbb{R}^{n \times d}$ for $i \in [s]$. Let s be the largest number of non-zero entries in a column of \mathbf{S} . Then \mathbf{S} can be decomposed such that $\mathbf{S} = \frac{1}{\sqrt{s}} \sum_{i \in [s]} \mathbf{S}_i$, where

$\mathbf{S}_i \in \{-1, 0, 1\}^{m \times n}$ is a matrix with only one non-zero entry per column, and the assignment of non-zero entries to indices i is done uniformly at random. T_S outputs the sum of matrix products $\frac{1}{\sqrt{s}} \sum_{i \in [s]} \mathbf{S}_i \mathbf{A}_i$.

$$T_S(\mathbf{A}) = \frac{1}{\sqrt{s}} \sum_{i \in [s]} \mathbf{S}_i \mathbf{A}_i \quad (3)$$

We also define $T_S^{\text{dense}} : \mathbb{R}^{n \times dm} \rightarrow \mathbb{R}^{m \times d}$, where \mathbf{S} is a dense matrix rather than a sparse one, so we set $s = m$. Then \mathbf{S} can be decomposed such that $\mathbf{S} = \frac{1}{\sqrt{m}} \sum_{i=1}^m \mathbf{S}_i$, where $\mathbf{S}_i \in \{-1, 0, 1\}^{m \times n}$ is a matrix with only one non-zero entry per column and exactly n/m non-zero entries per row.

Analysis function A . A takes a matrix in $\mathbb{R}^{m \times d}$, as well as any public values, and outputs some data analysis. Examples include low rank approximation and linear regression.

We now evaluate the noise distribution necessary to guarantee differential privacy in this model. We find that for sufficiently large n , we can apply the infinite divisibility of distributions to divide the noise into a number of pieces corresponding the number $(n - t')/2m$ of honest clients expected to contribute to any entry of a column in the resulting noisy matrix product. The concentration bound, formulated in Lemma 4.5 and used to bound this number of honest clients, contributes to the necessary value of δ . Composition theorems can be applied on the columns of the resulting matrix product to yield the final differential privacy guarantee. We will later see that adding suitable noise to data results in only a small error in the resulting error for regularized linear regression and low rank approximation.

We are now ready to state our privacy guarantees in the LTM.

Theorem 4.4. *Let $\epsilon \geq 0$, $\delta \in (0, 1)$, $m \in [n]$, $t' < n$, $s \in [m]$. Let $\mathbf{S} \sim \mathcal{D}_{m,n,s}^{\text{sketch}}$ with s non-zero entries per column. Also let noise g_i be sampled from a symmetric distribution D such that $\sum_{i \in [(n-s-t')/(2m)]} g_i = G$. G follows a distribution D' that defines an additive noise mechanism that satisfies $(\epsilon/(sd), \delta/(sd) - m \exp(-(n-s-t')/8m))$ -DP for function $f(\mathbf{A}) = (\mathbf{S}\mathbf{A})_j + \mathbf{n}$ for $\mathbf{n} \sim D'^m$ on input $\mathbf{A} \in \mathbb{R}^{n \times d}$, where the subscript refers to one column of $\mathbf{S}\mathbf{A}$. Then as long as input values are bounded above by η , the sparse sketching mechanism (R_D, T_S, A) , as defined by equations 2 and 3 and for any A , is (ϵ, δ) -differentially private in the trusted computation model for differential privacy with t' corrupt clients.*

PROOF. We begin by proving a helper lemma below.

Lemma 4.5. *Let $\gamma \in (0, 1)$, and let $\mathbf{S} \sim \mathcal{D}_{m,n}^{\text{sketch}}$. Then the probability that \mathbf{S} has at least $(1 - \gamma)\frac{n}{m}$ and at most $(1 + \gamma)\frac{n}{m}$ non-zero entries in every row is bounded by;*

$$\Pr\left(\exists i \in [m] : \left|X_i - \frac{n}{m}\right| > \gamma \cdot \frac{n}{m}\right) < 2m \exp\left(-\frac{\gamma^2 n}{2m}\right),$$

where X_i denotes the number of non-zero entries in row i .

PROOF. We only give a proof of the lower bound, as a proof of the upper bound is completely analogous. Let X_{ij} denote the indicator random variable that is 1 if \mathbf{S}_{ij} is non-zero and 0 otherwise. Note that each entry of \mathbf{S} can be thought of as an independent Bernoulli random variable X_{ij} , and the number of non-zero entries in a row is the sum of n of these random variables. Fixing a row $k \in [m]$, we

can use a Chernoff bound for sums of Bernoulli random variables to get a concentration bound around the expected number of non-zero entries $X_k = \sum_{j=1}^n X_{kj}$ in row k . In order to do this, we first need the expected value of X_k :

$$\mathbb{E}[X_k] = \sum_{j=1}^n \Pr(X_{kj} = 1) = \frac{n}{m}.$$

Thus a Chernoff bound gives us that

$$\Pr\left(X_k < (1 - \gamma) \frac{n}{m}\right) < \exp\left(-\frac{\gamma^2 n}{2m}\right).$$

Applying a union bound over all rows of \mathbf{S} then gives us exactly what we were to prove;

$$\begin{aligned} & \Pr\left(\exists i \in [m] : X_i < (1 - \gamma) \frac{n}{m}\right) \\ & \leq \sum_{k=1}^m \Pr\left(X_k < (1 - \gamma) \frac{n}{m}\right) \\ & < m \exp\left(-\frac{\gamma^2 n}{2m}\right). \end{aligned}$$

□

To prove Theorem 4.4, we first prove a simpler version of the theorem below.

Lemma 4.6. *Let $\varepsilon \geq 0$, $\delta \in (0, 1)$, $m \in [n]$, $t' < n$.*

Let $\mathbf{S} \sim \mathcal{D}_{m,n}^{\text{sketch}}$ with one non-zero entry per column. Also let $g_i \sim D$ be samples from a distribution such that $\sum_{i \in [(n-t')/(2m)]} g_i = G$. G follows a distribution D' that defines an additive noise mechanism that satisfies $(\varepsilon/d, \delta/d - m \exp(-(n-t')/8m))$ -DP for function $f(\mathbf{A}) = (\mathbf{S}\mathbf{A})_j + \mathbf{n}$ with $\mathbf{n} \sim D'^m$ on input $\mathbf{A} \in \mathbb{R}^{n \times d}$, where the subscript refers to one column of $\mathbf{S}\mathbf{A}$. Then as long as input values are bounded above by η , tuple of algorithms (R_D, T_S, A) , as defined by equations 2 and 3 and for any A , is (ε, δ) -differentially private in the trusted computation model for differential privacy with t' corrupt clients.

PROOF OF LEMMA 4.6. Let ε, δ, R_D and T_S be given as described in the theorem. If $\delta/d \leq m \exp(-(n-t')/8m)$, then the R_D outputs 0^d , which is entirely independent of the input and is therefore (ε, δ) differentially private.

Otherwise, $\mathcal{M}(\mathbf{A}) = T_S(R_D(\mathbf{a}_1), \dots, R_D(\mathbf{a}_n)) = \mathbf{S}(\mathbf{A} + \mathbf{G})$ where \mathbf{a}_i denotes row i of $\mathbf{A} \in \mathbb{R}^{n \times d}$ and $\mathbf{G} \leftarrow D^{n \times d}$, and we now prove that this is (ε, δ) differentially private.

Note that $\mathbf{S}_i(\mathbf{A} + \mathbf{G})$ is a d -dimensional vector, when \mathbf{S}_i refers to the i -th row of \mathbf{S} . If $(\mathbf{S}_i(\mathbf{A} + \mathbf{G}))_j$ is $(\varepsilon/d, \delta/d)$ -DP, then sequential composition (Lemma 3.5) gives that \mathcal{M} is (ε, δ) differentially private.

Now consider $d = 1$ and let mechanism $\mathcal{M}_1 : \mathbb{R}^n \rightarrow \mathbb{R}^m$ be defined as $\mathcal{M}_1(\mathbf{a}) = \mathbf{S}(\mathbf{a} + \mathbf{g})$ for $\mathbf{a} \in \mathbb{R}^n$, with $\mathbf{g} \leftarrow D^n$.

\mathbf{S} is sampled such that it has only one non-zero entry per column; thus, the columns of \mathbf{S} are orthogonal. This means that the entries of $(\mathbf{S}(\mathbf{a} + \mathbf{g}))_j$ and $(\mathbf{S}(\mathbf{a} + \mathbf{g}))_{j'}$ for $j \neq j'$, have disjoint support. Therefore, the privacy guarantee can be analyzed independently for each entry of $\mathbf{S}(\mathbf{a} + \mathbf{g})$.

Denote by E the event that the support of $\mathbf{S}(\mathbf{a} + \mathbf{g})_j$ has at least $\frac{n-t'}{2m}$ uncorrupted clients, whose set of indices we denote N . We can formulate $\mathbf{S}(\mathbf{a} + \mathbf{g})_j = (\mathbf{S}\mathbf{a})_j + (\mathbf{S}\mathbf{g})_j = (\mathbf{S}\mathbf{a})_j + \sum_{k \in [n]} \mathbf{S}_{j,k} \mathbf{g}_k = (\mathbf{S}\mathbf{a})_j + \sum_{k \in N} \mathbf{S}_{j,k} \mathbf{g}_k + \sum_{k \in [n] \setminus N} \mathbf{S}_{j,k} \mathbf{g}_k$. Also note that $\sum_{k \in N} \mathbf{S}_{j,k} \mathbf{g}_k$

contains at least $\frac{n-t'}{2m}$ terms in event E , the sum of which suffice for guaranteeing DP.

Due to the fact that D is symmetric, $-1 \cdot \mathbf{g}_k$ follows the same distribution as \mathbf{g}_k , and $\sum_{k \in N} \mathbf{S}_{j,k} \mathbf{g}_k$ follows the same distribution as $\sum_{k \in N} \mathbf{g}_k$. Due to the infinite divisibility of D' , the sum of the first $\frac{n-t'}{2m}$ terms of $\sum_{k \in N} \mathbf{g}_k$ are samples from D' .

Then in event E , \mathcal{M}_1 is $(\varepsilon, \delta - m \exp(-(n-t')/8m))$ differentially private using the additive noise mechanism that adds noise from D' . Setting $\gamma = 1/2$, Lemma 4.5 gives that E occurs with probability at least $1 - m \exp(-\frac{n-t'}{8m})$. Since event E does not occur with probability at most $m \exp(-(n-t')/8m)$, \mathcal{M}_1 is $(\varepsilon, \delta - m \exp(-(n-t')/8m))$ differentially private except with probability $m \exp(-(n-t')/8m)$. Then \mathcal{M}_1 is (ε, δ) -differentially private. □

Let ε, δ, R_D and T_S be given as described in the theorem. If $\delta/d \leq m \exp(-(n-s-t')/8m)$, then the R_D outputs 0^d , which is entirely independent of the input and is therefore (ε, δ) differentially private.

Otherwise, $\mathcal{M}(\mathbf{A}) = T_S(R_D(\mathbf{a}_1), \dots, R_D(\mathbf{a}_n)) = \frac{1}{\sqrt{s}} \sum_{i \in [s]} \mathbf{S}_i(\mathbf{A} + \mathbf{G}_i)$ where \mathbf{a}_i denotes row i of $\mathbf{A} \in \mathbb{R}^{n \times d}$ and $\mathbf{G}_i \leftarrow D^{n \times d}$. Here, \mathbf{S} has been decomposed such that $\mathbf{S} = \frac{1}{\sqrt{s}} \sum_i \mathbf{S}_i$, where $\mathbf{S}_i \in \{-1, 0, 1\}^{m \times n}$ is a matrix with only one non-zero entry per column.

We argue $(\varepsilon/s, \delta/s)$ -DP based on Lemma 4.6. Note that the non-zero entry per column in each \mathbf{S}_i is not sampled uniformly at random from all possible rows; rather, for \mathbf{S}_i is sampled uniformly at random from the remaining $n - i$ rows (removing the one that was already chosen to be non-zero, sampling without replacement). Therefore, we replace n with $n - s$ when applying Lemma 4.6.

By sequential composition, releasing $\mathbf{S}_i(\mathbf{A} + \mathbf{G}_i)$ for all $i \in [s]$ satisfies (ε, δ) -DP, and thus by post-processing, the scaled sum of these is also (ε, δ) -DP. □

One additive mechanism to which this can easily be applied is the Gaussian mechanism. To do so, D is a normal distribution with variance described in the following corollary.

Corollary 4.7. *Let $\varepsilon \geq 0$, $\delta \in (0, 1)$, $t' < n$, $m \in [n]$, $s \in [n]$ and D be the Gaussian distribution $\mathcal{N}(0, \sigma^2)$ with:*

$$\sigma^2 = \frac{4s^3 \eta^2 \ln(1.25s / (\delta/d - m \exp(-\frac{n-s-t'}{8m}))) m d^2}{\varepsilon^2 (n - s - t')}$$

Let $\mathbf{S} \sim \mathcal{D}_{m,n,s}^{\text{sketch}}$. Then as long as input values are bounded above by η , tuple of algorithms (R_D, T_S, A) , as defined by equations 2 and 3 and for any A , is (ε, δ) -differentially private in the trusted computation model for differential privacy with t' corrupt clients.

PROOF. We apply Theorem 4.4, setting D to a Gaussian distribution with the variance specified in the corollary. First note that zero-centered Gaussians are symmetric around the origin. Due to infinite divisibility of Gaussians, $\sum_{i \in [(n-s-t')/(2m)]} g_i = G$ when $g_i \sim D$ is Gaussian distributed with variance $\sum_{i \in [(n-s-t')/(2m)]} \sigma^2 = (n-s-t')\sigma^2/(2m)$

Notice that the ℓ_2 -sensitivity of algorithm T_S is $\Delta_2 T_S = \max_{A, A'} \|\mathbf{SA} - \mathbf{SA}'\|_2 = 2\eta\sqrt{s}$, where $A, A' \in \mathbb{R}^{n \times d}$ differ in a single row, and every entry of A, A' is bounded above by η .

Based on the Gaussian mechanism (Lemma 3.3), the additive noise mechanism adding Gaussian noise satisfies $(\epsilon/(sd), \delta/(sd) - m \exp(-(n-s-t')/8m))$ -DP for function $f(A) = (\mathbf{SA})_j + \mathbf{n}$ with $\mathbf{n} \sim \text{Lap}(0, 2\eta m^2 d/\epsilon)^m$ on input $A \in \mathbb{R}^{n \times d}$, where the subscript refers to one column of \mathbf{SA} . \square

If we would like to obtain a result for mechanisms with $\delta = 0$, we can instead use R_D^{dense} and use a dense sketching matrix $S \in \{-1, 1\}^{m \times n}$.

Theorem 4.8. *Let $\epsilon \geq 0, \delta \in [0, 1], m \in [n], t' < n, s \in [m]$. Let $S \sim \{-1, 1\}^{m \times n}$ be a dense, uniformly sampled matrix. Also let R add noise g_i sampled from a symmetric distribution D such that $\sum_{i \in [n/m-t']} g_i = G$. G follows a distribution D' that defines an additive noise mechanism that satisfies $(\epsilon/(md), \delta/(md))$ -DP for function $f(A) = (\mathbf{SA})_j + \mathbf{n}$ for $\mathbf{n} \sim D'^m$ on input $A \in \mathbb{R}^{n \times d}$, where the subscript refers to one column of \mathbf{SA} . Then as long as input values are bounded above by η , dense sketching mechanism $(R_D^{\text{dense}}, T_S^{\text{dense}}, A)$, as defined by equations 1 and 3 and for any A , is (ϵ, δ) -differentially private in the LTM for differential privacy with t' corrupt clients.*

PROOF. We again begin by proving a simplified version of the theorem first.

Lemma 4.9. *Let $\epsilon \geq 0, \delta \in [0, 1], m \in [n], t' < n$. Let $S \in \{-1, 0, 1\}^{m \times n}$ with one non-zero entry per column and n/m non-zero entries per row. Also let noise g_i be sampled from a symmetric distribution D such that $\sum_{i \in [(n)/m-t']} g_i = G$. G follows a distribution D' that defines an additive noise mechanism that satisfies $(\epsilon/d, \delta/d)$ -DP for function $f(A) = (\mathbf{SA})_j + \mathbf{n}$ with $\mathbf{n} \sim D'^m$ on input $A \in \mathbb{R}^{n \times d}$, where the subscript refers to one column of \mathbf{SA} . Then as long as input values are bounded above by η , tuple of algorithms $(R_D^{\text{dense}}, T_S, A)$, as defined by equations 1 and 3 and for any A , is (ϵ, δ) -differentially private in the trusted computation model for differential privacy with t' .*

PROOF OF LEMMA 4.9. Let $\epsilon, \delta, R_D^{\text{dense}}$ and T_S be given as described in the theorem.

We show that $\mathcal{M}(A) = T_S(R(\mathbf{a}_1), \dots, R(\mathbf{a}_n)) = \mathbf{S}(A + \mathbf{G})$ where \mathbf{a}_i denotes row i of $A \in \mathbb{R}^{n \times d}$ and $\mathbf{G} \leftarrow D^{n \times d}$, is (ϵ, δ) differentially private.

Note that $\mathbf{S}_i(A + \mathbf{G})$ is a d -dimensional vector, when \mathbf{S}_i refers to the i -th row of \mathbf{S} . If $(\mathbf{S}_i(A + \mathbf{G}))_j$ is $(\epsilon/d, \delta/d)$ -DP, then sequential composition (Lemma 3.5) gives that \mathcal{M} is (ϵ, δ) differentially private.

Consider $d = 1$ and let mechanism $\mathcal{M}_1 : \mathbb{R}^n \rightarrow \mathbb{R}^m$ be defined as $\mathcal{M}_1(\mathbf{a}) = \mathbf{S}(\mathbf{a} + \mathbf{g})$ for $\mathbf{a} \in \mathbb{R}^n$, with $\mathbf{g} \leftarrow D^n$.

\mathbf{S} is sampled such that it has only one non-zero entry per column; thus, the columns of \mathbf{S} are orthogonal. This means that the entries of $(\mathbf{S}(\mathbf{a} + \mathbf{g}))_j$ and $(\mathbf{S}(\mathbf{a} + \mathbf{g}))_{j'}$ for $j \neq j'$, have disjoint support. Therefore, the privacy guarantee can be analyzed independently for each entry of $\mathbf{S}(\mathbf{a} + \mathbf{g})$.

Note that the support of $\mathbf{S}(\mathbf{a} + \mathbf{g})_j$ has at least $\frac{n}{m} - t'$ uncorrupted clients, whose set of indices we denote N . We can formulate $\mathbf{S}(\mathbf{a} + \mathbf{g})_j = (\mathbf{S}\mathbf{a})_j + (\mathbf{S}\mathbf{g})_j = (\mathbf{S}\mathbf{a})_j + \sum_{k \in [n]} \mathbf{S}_{j,k} \mathbf{g}_k = (\mathbf{S}\mathbf{a})_j + \sum_{k \in N} \mathbf{S}_{j,k} \mathbf{g}_k + \sum_{k \in [n] \setminus N} \mathbf{S}_{j,k} \mathbf{g}_k$.

Due to the fact that D is symmetric, $-1 \cdot \mathbf{g}_k$ follows the same distribution as \mathbf{g}_k , and $\sum_{k \in N} \mathbf{S}_{j,k} \mathbf{g}_k$ follows the same distribution as $\sum_{k \in N} \mathbf{g}_k$. Due to the infinite divisibility of D' , the sum of the first $\frac{n}{m} - t'$ terms of $\sum_{k \in N} \mathbf{g}_k$ are samples from D' .

Then \mathcal{M}_1 is (ϵ, δ) -differentially private. \square

Let $\epsilon, \delta, R_D^{\text{dense}}$ and T_S^{dense} be given as described in the theorem.

We know $\mathcal{M}(A) = T_S(R_D(\mathbf{a}_1), \dots, R_D(\mathbf{a}_n)) = \frac{1}{\sqrt{m}} \sum_{i \in [m]} \mathbf{S}_i(A + \mathbf{G}_i)$ where $\mathbf{G}_i \leftarrow D^{n \times d}$. Recall from Equation 3 that \mathbf{S} has been decomposed such that $\mathbf{S} = \frac{1}{\sqrt{m}} \sum_i^m \mathbf{S}_i$, where $\mathbf{S}_i \in \{-1, 0, 1\}^{m \times n}$ is a matrix with only one non-zero entry per column and n/m non-zero entries per row.

We argue $(\epsilon/s, \delta/s)$ -DP based on Lemma 4.9.

By sequential composition, releasing $\mathbf{S}_i(A + \mathbf{G}_i)$ for all $i \in [s]$ satisfies (ϵ, δ) -DP, and thus by post-processing, the scaled sum of these is also (ϵ, δ) -DP. \square

To instantiate this second theorem, D can be chosen as the distribution formulated as the difference of two samples of the gamma distribution.

Corollary 4.10. *Let $\epsilon \geq 0, t' < n, m \in [n], s \in [n]$, and let D be the distribution formulated as the difference of two samples of gamma distribution $\Gamma(\frac{1}{n/m-t'}, b)$ with parameter*

$$b = 2\eta m^2 d/\epsilon$$

Let $S \sim \{-1, 1\}^{m \times n}$ sampled uniformly. Then as long as input values are bounded above by η , $(R_D^{\text{dense}}, T_S, A)$ is ϵ -differentially private in the trusted computation model for differential privacy with t' corrupt clients C' .

PROOF. We apply Theorem 4.8, setting D to the difference between two samples of the gamma distribution $\Gamma(\frac{1}{n/m-t'}, 2\eta m^2 d/\epsilon)$. Note that the resulting distribution is centered at 0 and symmetric. Due to infinite divisibility of Laplace distributions, $\sum_{i \in [n/m-t']} g_i = G$ when $\mathbf{g}_i \sim D$ is Laplace distributed with mean zero and scale parameter $b = 2\eta m^2 d/\epsilon$.

Notice that the ℓ_1 -sensitivity of algorithm T_S is $\Delta_1 T_S = \max_{A, A'} \|\mathbf{SA}_j - \mathbf{SA}'_j\|_1 = 2\eta m$, where $A, A' \in \mathbb{R}^{n \times d}$ differ in a single row, and every entry of A, A' is bounded above by η .

Based on the Laplace mechanism (Lemma 3.4), the additive noise mechanism adding Laplace noise satisfies $\epsilon/(dm)$ -DP for function $f(A) = (\mathbf{SA})_j + \mathbf{n}$ with $\mathbf{n} \sim \text{Lap}(0, 2\eta m^2 d/\epsilon)^m$ on input $A \in \mathbb{R}^{n \times d}$, where the subscript refers to one column of \mathbf{SA} . \square

Corollary 4.11 below follows directly from Theorem 4.4 and Lemma 4.3.

Corollary 4.11. *Let (R_D, T_S, A) be the tuple of algorithms above. If T_S is computed correctly using an MPC protocol Π_S with perfect security tolerating t semi-honest corruptions, then the tuple of algorithms (R_D, Π_S, A) is (ϵ, δ) -differentially private with t' corrupt clients.*

4.3 Cryptographic Assumptions and Relations to the Shuffle Model

The LTM can be securely distributed using simple cryptographic techniques for *secure multiparty computation* (MPC), such as *linear secret-sharing* (LSS). MPC ensures that no adversary corrupting all but one server learns more than the computation’s output. [27] first explored combining differential privacy and MPC, focusing on distributed noise generation requiring server interaction. [19] establish lower bounds for non-interactive multi-server mechanisms, while [25] propose an interactive MPC protocol for selection in distributed trust models.

The linear transforms can be non-interactively computed locally by servers and are also known to all participating parties, that is, once initiated, the output of the protocol is fully deterministic. Linear transforms can be computed non-interactively by servers and are publicly known, making the protocol fully deterministic once initiated. This enables implementing the LTM with multiple central servers under the minimal assumption that at least one server remains honest. Using LSS has key advantages: clients send a single message per server proportional to the input size and sparsity s of the transformation matrix S , servers communicate only with the data analyst, and the total workload scales only with the number of servers and matrix sparsity. Since LSS supports information-theoretic security, our system remains secure even against quantum adversaries.

We require this secret sharing scheme to be information-theoretically secure. We use an additive secret sharing scheme, such that clients C_1, \dots, C_n generate k shares of their noisy inputs x^1, \dots, x^n , where k is the number of servers that compute the linear transformation. We use $[x^i]$ to denote a secret sharing of some client C_i ’s noisy input x^i for some $i \in [n]$. For some field \mathbb{F} of size p and some prime number p , $[x^i]$ consists of shares $x_1^i, \dots, x_k^i \in \mathbb{F}$ such that $\sum_{j=1}^k x_j^i = x^i$. To split a secret into k shares, a client can sample $k-1$ random field elements x_1^i, \dots, x_{k-1}^i and compute the last secret share as $x^i - \sum_{j=1}^{k-1} x_j^i$. Therefore, it also seems intuitive that an adversary that sees all but one of the shares knows nothing about the input. For every $j \in [k]$, server S_j receives $\{x_i^j\}_{i \in [n]}$ from the respective parties. We define these shares such that the security of our distributed protocol does not rely on any computational assumption.

Since the linear transformation is public, servers can apply the transformation locally to their secret shares without any need to communicate with each other. Each server then reveals their shares of the resulting linear transformation, such that the linear sketch can be revealed in the clear. For more details on MPC based on secret sharing, see e.g., [24].

Secure aggregation and the shuffle model are two instantiations of intermediate trust models for differential privacy. Secure aggregation [2, 15, 37, 57, 67] is a special case of the LTM, where the linear transformation applied is a sum, which is useful in federated learning. The shuffle model [12, 18, 31] can be distributed among multiple servers using cryptographic protocols such as mixnets.

However, implementing shuffles is more complex than linear transformations, requiring servers to communicate over a chain

(increasing latency), computationally intensive public-key cryptography, which in turns requires computational assumptions. In contrast, LSS enables efficient, unconditional security and deterministic computation.¹

5 Numerical Linear Algebra in LTM

In this section we give the utility guarantees of our mechanism. We start with giving generic distortion bounds relating the spectrum of the noisy matrix to the spectrum of the original non-private matrix. Parameterizations of this mechanism depend on the underlying class of subspace embeddings. In this utility analysis, we limit ourselves to OSNAPs with m and s chosen according to [22]; other trade-offs are possible and can be found in Section 3. Applications to specific problems such as regression and low rank approximation are given towards the end of the section.

Here we present the formal utility statements for ridge regression and low rank approximation. We begin by introducing a useful helper lemma.

Lemma 5.1. *Let $G \in \mathbb{R}^{n \times d}$ such that all entries in G are independently sampled from a normal distribution with mean 0 and variance at most σ^2 . Further, let V be a set of d -dimensional vectors lying in a k -dimensional subspace. Then with probability at least $1 - \beta$ for some absolute constant η*

$$\sup_{x \in V} \|Gx\|_2^2 \leq n \cdot \sigma^2 \cdot \|x\|^2 + \eta \cdot \sigma^2 \cdot \|x\|^2 \cdot (\sqrt{(k + \log 1/\beta) \cdot n} + (k + \log 1/\beta)).$$

Moreover, with probability at least $1 - \beta$

$$\|G\|_F^2 \leq \eta \cdot \sigma^2 \cdot n \cdot d \cdot \log 1/\beta.$$

PROOF. Denote by G_i the i th row of G . We start by observing that $G_i^T x$ is Gaussian distributed with mean 0 and variance $\sigma^2 \cdot \|x\|^2$. Thus, we focus our attention on controlling $\frac{\|Gx\|_2^2}{\sigma^2 \cdot \|x\|^2}$, that is, we assume that x is a unit vector and that the entries of G are standard normal Gaussian random variables. The overall claim then follows by rescaling. Consider an ε -net N_ε of V , that is, for every $x \in V$ with unit norm there exists a vector $x' \in N_\varepsilon$ with $\|x - x'\| \leq \varepsilon$. Such nets exist with $|N_\varepsilon| \leq \exp(\eta \cdot k \log 1/\varepsilon)$ for some absolute constant η , see [61]. Suppose $\varepsilon = 1/4$. Using concentration bounds for sums of Gaussians (see for example Lemma 1 of [50]), we then have for any $\|x'\|$ with probability $1 - \beta$

$$\begin{aligned} & \mathbb{P}[\exists x' \in N_\varepsilon : \|Gx'\|^2 - \mathbb{E}[\|Gx'\|^2] \\ & \geq 2(\sqrt{(\log N_\varepsilon + \log 1/\beta) \cdot n} + (\log N_\varepsilon + \log 1/\beta))] \\ & \leq |N_\varepsilon| \cdot \exp(-\eta \cdot k + \log 1/\beta) \leq \beta. \end{aligned}$$

for some absolute constant η .

We now extend this argument to all vectors, using an argument from [4] (Lemma 4 of that reference). Let U be an orthogonal basis of V . Then our goal is to control $\|Gx\|^2 = x^T U^T G^T G U x$. Define the matrix $B := U^T G^T G U - n \cdot I$, where I is the identity matrix. Note

¹The linear transformations we use are random but public and sampled before the computation.

that $\mathbb{E}[\mathbf{U}^T \mathbf{G}^T \mathbf{G} \mathbf{U}] = n \cdot \mathbf{I}$ and that $\|\mathbf{B}\mathbf{x}\|$ is the deviation of $\|\mathbf{G}\mathbf{x}\|$ around its expectation. Let $\|\mathbf{B}\|_{op} := \sup_{\mathbf{x} \in V} \|\mathbf{B}\mathbf{x}\|$.

$$\begin{aligned} \|\mathbf{B}\|_{op} &= \langle \mathbf{B}\mathbf{x}, \mathbf{x} \rangle = \langle \mathbf{B}\mathbf{x}', \mathbf{x}' \rangle + \langle \mathbf{B}(\mathbf{x}' + \mathbf{x}), \mathbf{x}' - \mathbf{x} \rangle \\ &\leq 2(\sqrt{(\log N_Y + \log 1/\beta) \cdot n} + (\log N_Y + \log 1/\beta)) \\ &\quad + \|\mathbf{B}\|_{op} \cdot \|\mathbf{x}' + \mathbf{x}\| \|\mathbf{x}' - \mathbf{x}\| \\ &\leq 2(\sqrt{(\log N_Y + \log 1/\beta) \cdot n} + (\log N_Y + \log 1/\beta)) \\ &\quad + 2\gamma \cdot \|\mathbf{B}\|_{op} \\ &\leq 2(\sqrt{(\log N_Y + \log 1/\beta) \cdot n} + (\log N_Y + \log 1/\beta)) \\ &\quad + 1/2 \cdot \|\mathbf{B}\|_{op} \end{aligned}$$

Rearranging implies that $\|\mathbf{B}\|_{op} \leq 4(\sqrt{(\log N_Y + \log 1/\beta) \cdot n} + (\log N_Y + \log 1/\beta))$. The first claim now follows by rescaling η .

For the second claim, we observe that $\|\mathbf{G}\|_F^2$ has a Gaussian distribution with mean 0 and variance $n \cdot d \cdot \sigma^2$. The same concentration inequality we applied above also implies that the probability that $\|\mathbf{G}\|_F$ exceeds $2\sqrt{\sigma^2 \cdot n \cdot d \cdot \log 1/\beta}$ is at most $1 - \beta$. \square

We will use the following spectral bounds for both linear regression and low rank approximation. We believe that there may be further applications and that the bounds themselves are therefore of independent interest.

Lemma 5.2. *Let $\mathbf{S} \in \frac{1}{\sqrt{s}} \cdot \{-1, 0, 1\}^{m \times n}$ be an (α, β, m, s) OSNAP with $\mathbf{S} = \frac{1}{\sqrt{s}} \sum_{i \in [s]} \mathbf{S}_i$, where $\mathbf{S}_i \in \{-1, 0, 1\}^{n \times m}$ has exactly one non-zero entry per row. Let $\mathbf{G} = \sum_{i \in [s]} \mathbf{S}_i \mathbf{G}_i$ where every matrix $\mathbf{G}_i \in \mathbb{R}^{n \times d}$ has independent Gaussian entries $\mathcal{N}(0, \sigma^2)$. Further, let V be a set of d -dimensional vectors lying in a k -dimensional subspace. Then with probability at least $1 - \beta$ for some absolute constant η*

$$\sup_{\mathbf{x} \in V} \frac{1}{s} \|\mathbf{G}\mathbf{x}\|_2^2 \leq \eta \cdot 2 \frac{n}{m} \sigma^2 \cdot \|\mathbf{x}\|^2 \cdot (\sqrt{(k + \log 1/\beta) \cdot m} + (k + \log 1/\beta)).$$

and

$$\frac{1}{s} \|\mathbf{G}\|_F^2 \leq \eta \cdot 2n\sigma^2 \cdot d \cdot \log 1/\beta$$

PROOF. We first argue that \mathbf{G} is Gaussian distributed. Each \mathbf{G}_i has independent Gaussian entries and multiplying a Gaussian with a random Rademacher does not change the distribution. Therefore, the entries of $\mathbf{S}_i \mathbf{G}_i$ are likewise Gaussian distributed, with mean 0 and variance at most $(1 + \gamma) \frac{n}{m} \sigma^2 \leq 2 \frac{n}{m} \sigma^2$ due to Lemma 4.5. Concluding, the variance of the entries of $\mathbf{G} = \sum_{i \in [s]} \mathbf{S}_i \mathbf{G}_i$ is therefore at most $2s \frac{n}{m} \sigma^2$. Applying Lemma 5.1, we then have with probability $1 - \beta$

$$\begin{aligned} \sup_{\mathbf{x} \in V} \|\mathbf{G}\mathbf{x}\|_2^2 &\leq 2 \frac{n}{m} \sigma^2 \cdot \|\mathbf{x}\|^2 + \eta \cdot 2s \frac{n}{m} \sigma^2 \cdot \|\mathbf{x}\|^2 \\ &\cdot (\sqrt{(k + \log 1/\beta) \cdot m} + (k + \log 1/\beta)). \end{aligned}$$

and

$$\|\mathbf{G}\|_F^2 \leq \eta \cdot 2sn\sigma^2 \cdot d \cdot \log 1/\beta$$

as desired \square

Related, though slightly weaker bounds can also be derived for the Gamma/Laplace mechanism. While it is possible to derive bounds for sparse Rademacher sketches, we focus on the dense constructions as these yield ϵ differentially private mechanisms (as opposed to (ϵ, δ) differentially private mechanisms).

Lemma 5.3. *Let $\mathbf{S} \in \frac{1}{\sqrt{s}} \cdot \{-1, 0, 1\}^{m \times n}$ be an (α, β, m, m) OSNAP with $\mathbf{S} = \frac{1}{\sqrt{m}} \sum_{i \in [s]} \mathbf{S}_i$, where $\mathbf{S}_i \in \{-1, 0, 1\}^{m \times n}$ has exactly one non-zero entry per row and n/m non-zero entries per column. Let $\mathbf{L} = \frac{1}{\sqrt{m}} \sum_{i \in [m]} \mathbf{S}_i \Gamma_i$ where every matrix $\Gamma_i \in \mathbb{R}^{n \times d}$ has independent entries $X_i - Y_i$, where $X_i, Y_i \sim \Gamma(m/n, b)$. Then with probability at least $1 - \beta$ for some absolute constant η and*

$$\|\mathbf{L}\|_F^2 \leq \eta dm^2 b^2 \log^2 \frac{m}{\beta}$$

PROOF. We have $\|\mathbf{L}\|_F \leq \sqrt{m} \cdot \|\max_{i \in [m]} \mathbf{S}_i \Gamma_i\|_F$, thus it is sufficient to give a high probability bound on $\|\mathbf{S}_i \Gamma_i\|_F$.

The entries of $\mathbf{S}_i \Gamma_i$ are Laplace distributed entries with mean 0 and scale parameter b by infinite divisibility of a Laplace random variable into Γ distributed random variables. Thus the expected value of $\|\mathbf{S}_i \Gamma_i\|_F^2$ is $md \cdot 2b^2$.

To prove high concentration, let us consider the moments of the Laplace distribution. For a Laplace random variable L with mean 0 and scale b , we have $\mathbb{E}[L^{2\ell}] = (2\ell)! b^{2\ell}$. Then due to Hoelder's inequality

$$\begin{aligned} \mathbb{E}[\|\mathbf{S}_i \Gamma_i\|_F^{2\ell}] &= \mathbb{E}\left[\left(\sum_{j=1}^{md} L_j^2\right)^\ell\right] \\ &\leq \mathbb{E}\left[\left(\sum_{j=1}^{md} L_j^{2\ell}\right)\right] \cdot \left(\sum_{j=1}^{md} 1^{\frac{\ell}{\ell-1}}\right)^{\ell-1} \\ &= (2\ell)! b^{2\ell} \cdot (md)^\ell \end{aligned}$$

Applying Markov's inequality, we then have for any $\gamma > 1$

$$\begin{aligned} &\mathbb{P}[\|\mathbf{S}_i \Gamma_i\|_F^{2\ell} > \gamma \cdot md \cdot 2b^{2\ell}] \\ &= \mathbb{P}\left[\|\mathbf{S}_i \Gamma_i\|_F^{2\ell} > \left(\gamma \cdot md \cdot 2b^{2\ell}\right)^\ell\right] \\ &\leq \frac{(2\ell)! b^{2\ell} (md)^\ell}{(\gamma \cdot md \cdot 2b^{2\ell})^\ell} = \frac{(2\ell)!}{(2\gamma)^\ell} \\ &\leq \frac{(2\ell)^{2\ell}}{(2\gamma e^2)^\ell} \sqrt{6\pi\ell} \end{aligned}$$

For $\gamma > 4\ell^2$ and $\ell = \log \frac{m}{\beta}$, this term is at most β/m . We can therefore conclude that with probability $1 - \beta$, $\max_{i \in [m]} \mathbf{S}_i \Gamma_i \leq 4 \log \frac{m}{\beta}$ and thus $\|\mathbf{L}\|_F^2 \leq \eta m^2 db^2 \log^2 \frac{m}{\beta}$, for some absolute constant η . \square

We first begin with our utility results for low rank approximation. Using the Gaussian mechanism, we obtain the following guarantee.

Theorem 5.4. *Let $\epsilon \geq 0$, $\delta \in (0, 1)$, $t' < n/2$. Let $\mathbf{S} \sim \mathcal{D}_{m,n}^{\text{sketch}}$. Define A_k as an algorithm that computes $\text{argmin}_{\mathbf{X} \text{ rank } k, \mathbf{X} \in \mathbb{R}^{n \times d}} \|\mathbf{A} - \mathbf{A}\mathbf{X}\mathbf{X}^T\|_F^2$. An instantiation of the trusted computation model (R_D, T_S, A_k) specified in Equations 2 and 3, using noise distribution $D \sim \mathcal{N}(0, \sigma^2)$ according to Corollary 4.7, computes an orthogonal projection \mathbf{X}' that with probability $1 - \beta$ achieves a multiplicative error $(1 + O(\alpha_S))$*

and additive error $O\left(\frac{kd^{3/2}}{\alpha_S^2} \log^{3/2}(1/\beta)\epsilon^{-1} \log^{1/2}(1/\delta)\right)$ for the low rank approximation problem.

PROOF. We will use the following inequality. The applications to vector and matrix norms are straightforward corollaries.

Lemma 5.5 (Generalized Triangle Inequality [9]). *For any two real numbers a, b and any $\alpha > 0$*

$$|a^2 - b^2| \leq \alpha \cdot a^2 + \left(1 + \frac{1}{\alpha}\right) \cdot (a - b)^2.$$

Finally, we require the following consequence of subspace embedding properties.

Lemma 5.6 (Paraphrased from Definition 1, Lemma 11, and Theorem 12 [23]). *S be drawn from an OSNAP distribution. Then for any matrix $A \in \mathbb{R}^{n \times d}$ and any rank k orthogonal matrix $X \in \mathbb{R}^{d \times k}$, we have*

$$\|SA - SAXX^T\|_F = (1 \pm \alpha) \cdot \|A - AXX^T\|_F.$$

Without noise, the low rank approximation outputs $X_{\text{OPT}} = \operatorname{argmin}_{X \text{ rank } k, \in \mathbb{R}^{n \times d}} \|A - AXX^T\|_F^2$, and after sketching with Gaussian noise of variance σ^2 outputs $X' = \operatorname{argmin}_{X \text{ rank } k, \in \mathbb{R}^{n \times d}} \|(A + G) - (A + G)XX^T\|_F^2$. Let X' be the matrix returned by the mechanism. We have

$$\begin{aligned} & \|A - AX'X'^T\|_F \\ & \leq (1 + \alpha_S) \sqrt{\frac{1}{s}} \left\| \sum_{i \in [s]} S_i A - S_i AX'X'^T \right\|_F \\ & \leq (1 + \alpha_S) \sqrt{\frac{1}{s}} \\ & \left\| \sum_{i \in [s]} S_i (A + G_i) - S_i (A + G_i) X' X'^T \right\|_F \\ & + (1 + \alpha_S) \sqrt{\frac{1}{s}} \left\| \sum_{i \in [s]} S_i G_i - S_i G_i X' X'^T \right\|_F \end{aligned}$$

where the first inequality follows from the subspace embedding property. By optimality of X' for the low rank approximation problem on $\sum_{i \in [s]} S_i (A + G_i)$, we then have

$$\begin{aligned} & \|A - AX'X'^T\|_F \\ & \leq (1 + \alpha_S) \sqrt{\frac{1}{s}} \\ & \cdot \left\| \sum_{i \in [s]} S_i (A + G_i) - S_i (A + G_i) X_{\text{OPT}} X_{\text{OPT}}^T \right\|_F \\ & + (1 + \alpha_S) \sqrt{\frac{1}{s}} \left\| \sum_{i \in [s]} S_i G_i - S_i G_i X' X'^T \right\|_F \\ & \leq (1 + \alpha_S)^2 \|A - AX_{\text{OPT}} X_{\text{OPT}}^T\|_F \\ & + (1 + \alpha_S) \sqrt{\frac{1}{s}} \left\| \sum_{i \in [s]} S_i G_i - S_i G_i X_{\text{OPT}} X_{\text{OPT}}^T \right\|_F \\ & + (1 + \alpha_S) \sqrt{\frac{1}{s}} \left\| \sum_{i \in [s]} S_i G_i - S_i G_i X' X'^T \right\|_F \end{aligned}$$

where the inequality follows from Lemma 5.6. Now notice that $\mathbb{I} - X_{\text{OPT}} X_{\text{OPT}}^T$ and $\mathbb{I} - X' X'^T$ are orthogonal projections, multiplying by which cannot increase the Frobenius norm of a matrix. Therefore, for $X = X'$ or $X = X_{\text{OPT}}$:

$$\begin{aligned} & \sqrt{\frac{1}{s}} \left\| \sum_{i \in [s]} S_i G_i - S_i G_i X X^T \right\|_F \\ & = \sqrt{\frac{1}{s}} \left\| \sum_{i \in [s]} S_i G_i (\mathbb{I} - X X^T) \right\|_F \leq \sqrt{\frac{1}{s}} \left\| \sum_{i \in [s]} S_i G_i \right\|_F \end{aligned}$$

Then:

$$\begin{aligned} & \|A - AX'X'^T\|_F \\ & \leq (1 + \alpha_S)^2 \|A - AX_{\text{OPT}} X_{\text{OPT}}^T\|_F \\ & + (1 + \alpha_S) 2 \frac{1}{s} \left\| \sum_{i \in [s]} S_i G_i \right\|_F \end{aligned} \quad (4)$$

Using Lemma 5.2, we have $\frac{1}{s} \left\| \sum_{i \in [s]} S_i G_i \right\|_F^2 \leq 2n \cdot \sigma^2 \cdot d \log(1/\beta)$. To simplify the error, observe that for the sketching matrix by [22] (see Table 2), we can set $s = \frac{\log k}{\alpha_S}$ and $m = \frac{k \log k}{\alpha_S^2} \log(1/\beta)$. Using the bound on the variance from Theorem 4.7 and plugging this in above, we can simplify the additive and multiplicative errors.

$$\begin{aligned}
& \| \mathbf{A} - \mathbf{A}\mathbf{X}'\mathbf{X}'^T \|_F^2 \\
& \leq (1 + \alpha_S)^2 \| \mathbf{A} - \mathbf{A}\mathbf{X}_{\text{OPT}}\mathbf{X}_{\text{OPT}}^T \|_F^2 \\
& \quad + \eta(1 + \alpha_S) \sqrt{\sigma^2 \cdot n \cdot d \cdot \log \frac{1}{\beta}} \\
& \leq (1 + 3\alpha_S) \| \mathbf{A} - \mathbf{A}\mathbf{X}_{\text{OPT}}\mathbf{X}_{\text{OPT}}^T \|_F^2 \\
& \quad + \tilde{O} \left(\frac{kd^{3/2}}{\alpha_S^3} \log^{3/2}(1/\beta) \varepsilon^{-1} \log^{1/2}(1/\delta) \right)
\end{aligned}$$

The stated bound follows by rescaling α_S by a factor of 3. \square

Using dense Rademacher sketches and the Gamma/Laplace mechanism, we instead obtain the following guarantee.

Theorem 5.7. *Let $\varepsilon \geq 0$, $\delta \in (0, 1)$, $t' < n/2$. Let $\mathbf{S} \sim \mathcal{D}_{m,n}^{\text{sketch}}$. Define A_k as an algorithm that computes $\text{argmin}_{\mathbf{X}} \text{rank } k, \varepsilon \in \mathbb{R}^{n \times d} \| \mathbf{A} - \mathbf{A}\mathbf{X}\mathbf{X}^T \|_F^2$. An instantiation of the trusted computation model $(R_D^{\text{dense}}, T_S^{\text{dense}}, A_k)$ specified in Equations 1 and 3, using noise distribution D as the difference between two independent Γ random variables parameterized by Corollary 4.10, computes an orthogonal projection \mathbf{X}' that with probability $1 - \beta$ with multiplicative error $(1 + O(\alpha_S))$ and additive error $O(\frac{k^3 d^{3/2}}{\alpha_S^6} \varepsilon^{-1} \log \frac{m}{\beta})$.*

PROOF. The proof follows the same line of reasoning as Theorem 5.4. The main difference is that upon reaching Equation 4, we must bound $\frac{1}{m} \left\| \sum_{i \in [m]} \mathbf{S}_i \mathbf{G}_i \right\|_F^2$ via Lemma 5.3 rather than using Lemma 5.2. Assuming the semi-trusted corruption model with at most t corruptions, each entry of \mathbf{G}_i consists of $X_j - Y_j$ where $X_j, Y_j \sim \Gamma(1/(n-t), b)$. Consequently, the entries of $\sum_{i \in [m]} \mathbf{S}_i \mathbf{G}_i$ are a sum of $m \cdot \lceil \frac{n}{n-t} \rceil$ matrices where each entry is either 0 or a Laplacian distributed random variable with mean 0 and scale b . Using Lemma 5.3, we therefore have with probability $1 - \beta$.

$$\begin{aligned}
& \frac{1}{m} \left\| \sum_{i \in [m]} \mathbf{S}_i \mathbf{G}_i \right\|_F^2 \\
& \in O(1) \cdot m^2 \cdot \frac{n}{n-t} db^2 \log^2 \frac{m \cdot \frac{n}{n-t}}{\beta} \\
& \in O(1) \cdot m^2 \cdot db^2 \log^2 \frac{m}{\beta},
\end{aligned}$$

where we used $\frac{n}{n-t} \leq 2$ by assumption. Since \mathbf{S} is a dense Rademacher matrix, we have $m = \frac{k + \log \beta^{-1}}{\alpha_S^2}$ [1, 77] and $s = m$, and $b = 2\eta m^2 d / \varepsilon$ following Corollary 4.10.

Thus we obtain, with probability $1 - \beta$, a $(1 + \alpha_S)$ multiplicative approximation using the same calculation as in Theorem 5.4 and a suitable rescaling, and an additive $O(\frac{k^3 d^{3/2}}{\alpha_S^6} \varepsilon^{-1} \log^2 \frac{k}{\alpha_S \beta})$ approximation. \square

For ridge regression, we achieve the following bound.

Theorem 5.8. *Let $\varepsilon \geq 0$, $\delta \in (0, 1)$, $t' < n$ and σ be chosen as described in Theorem 4.7. Let $\mathbf{S} \sim \mathcal{D}_{m,n}^{\text{sketch}}$. Define A_λ that performs linear regression on the sketched noisy inputs, outputting $\text{argmin}_{\mathbf{x}} \| \mathbf{A}\mathbf{x} - \mathbf{b} \|^2 + \lambda \| \mathbf{x} \|^2$, where the error is measured by $\min_{\mathbf{x}} \| \mathbf{A}\mathbf{x} - \mathbf{b} \|^2 + \lambda \| \mathbf{x} \|^2$.*

An instantiation of the LTM (R_D, T_S, A_λ) specified in Equations 2 and 3, using noise distribution $D \sim \mathcal{N}(0, \sigma^2)$ according to Corollary 4.7, which tolerates t' corrupted clients, computes linear regression parameters for any $\alpha_S > 0$ with probability $1 - \beta$ with multiplicative error $1 + \tilde{O}(\alpha_S + p + p^2)$ and additive error $\tilde{O}(p + p^2)$, where $p = \frac{d^3 \alpha_S^{-5} \log^5 \frac{1}{\beta} \cdot \varepsilon^{-2} \log \frac{1}{\delta}}{\lambda}$.

PROOF. Let ε, δ, R_D and T_S be given as in the theorem. We consider the output $A_{\text{reg}}(T_S(R_D(\mathbf{a}_1), \dots, R_D(\mathbf{a}_n)))$.

If $n \leq 8m \ln(dm/\delta) + t'$, then the R_D outputs 0^d , then the optimal $\mathbf{x}' = 0^d$, leading to error $\| \mathbf{A}\mathbf{x}' - \mathbf{b} \|^2 + \lambda \| \mathbf{x}' \|^2 = \| \mathbf{b} \|^2 \leq \eta' n \leq \eta' m \ln(dm/\delta) + t'$.

Otherwise, $\mathcal{M}(\mathbf{A}) = T_S(R_D(\mathbf{a}_1), \dots, R_D(\mathbf{a}_n)) = \frac{1}{\sqrt{s}} \sum_{i \in [s]} \mathbf{S}_i (\mathbf{A} + \mathbf{G}_i)$ where \mathbf{a}_i denotes row i of $\mathbf{A} \in \mathbb{R}^{n \times d}$ and $\mathbf{G} \leftarrow \mathcal{N}(0, \sigma^2)^{n \times d}$, is (ε, δ) differentially private.

To streamline the presentation, we give the analysis without corrupted clients. Adding corrupted clients merely changes the analysis along the same lines as Theorem 5.4.

We first control the terms $\| \sum_{i \in [s]} \mathbf{S}_i (\mathbf{G}_i \mathbf{x}' - \mathbf{g}_i) \|^2$ and $\| \sum_{i \in [s]} \mathbf{S}_i (\mathbf{G}_i \mathbf{x}_{\text{OPT}} - \mathbf{g}_i) \|^2$. Using Lemma 5.2 with an added coordinate of -1 to both \mathbf{x}' and \mathbf{x}_{OPT} , we get

$$\frac{1}{s} \left\| \sum_{i \in [s]} \mathbf{S}_i (\mathbf{G}_i \mathbf{x} - \mathbf{g}_i) \right\|^2 \leq \eta \cdot \sigma^2 nd \log \frac{1}{\beta} (\| \mathbf{x} \|^2 + 1) \quad (5)$$

for either $\mathbf{x} = \mathbf{x}_{\text{OPT}}$ or $\mathbf{x} = \mathbf{x}'$ and for a sufficiently large constant η . Then we have using Lemma 5.6

$$\begin{aligned}
& \| \mathbf{A}\mathbf{x}' - \mathbf{b} \|^2 + \lambda \| \mathbf{x}' \|^2 \\
& \leq (1 + \alpha_S) \| \mathbf{S}(\mathbf{A}\mathbf{x}' - \mathbf{b}) \|^2 + \lambda \| \mathbf{x}' \|^2 \\
& = (1 + \alpha_S) \frac{1}{s} \left\| \sum_{i \in [s]} \mathbf{S}_i (\mathbf{A}\mathbf{x}' - \mathbf{b}) \right\|^2 + \lambda \| \mathbf{x}' \|^2 \\
& = \lambda \| \mathbf{x}' \|^2 + (1 + \alpha_S) \frac{1}{s} \\
& \quad \cdot \left\| \sum_{i \in [s]} \mathbf{S}_i ((\mathbf{A} + \mathbf{G}_i - \mathbf{G}_i) \mathbf{x}' - \mathbf{b} + \mathbf{g}_i - \mathbf{g}_i) \right\|^2 \\
& \leq (1 + \alpha_S) \frac{1}{s} \left((1 + \alpha_S) \left\| \sum_{i \in [s]} \mathbf{S}_i ((\mathbf{A} + \mathbf{G}_i) \mathbf{x}' - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 \right. \\
& \quad \left. + \left(1 + \frac{1}{\alpha_S} \right) \left\| \sum_{i \in [s]} \mathbf{S}_i (\mathbf{G}_i \mathbf{x}' - \mathbf{g}_i) \right\|^2 \right) + \lambda \| \mathbf{x}' \|^2 \\
& \leq (1 + \alpha_S)^2 \cdot \left(\frac{1}{s} \left\| \sum_{i \in [s]} \mathbf{S}_i ((\mathbf{A} + \mathbf{G}_i) \mathbf{x}' - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 \right. \\
& \quad \left. + \lambda \cdot \| \mathbf{x}' \|^2 \right) + 2 \cdot \left(1 + \frac{1}{\alpha_S} \right) \eta \sigma^2 nd \log \frac{1}{\beta} (\| \mathbf{x}' \|^2 + 1) \quad (7)
\end{aligned}$$

where the second to last inequality follows by applying Lemma 5.5 and the final inequality follows from Equation 5. By optimality of \mathbf{x}' for the instance $\frac{1}{s} \left\| \sum_{i \in [s]} \mathbf{S}_i (\mathbf{A}\mathbf{x}' + \mathbf{G}_i - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 + \lambda \| \mathbf{x}' \|^2$,

we then have

$$\begin{aligned} & \frac{1}{s} \left\| \sum_{i \in [s]} S_i((\mathbf{A} + \mathbf{G}_i)\mathbf{x}' - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 + \lambda \cdot \|\mathbf{x}'\| \\ & \leq \frac{1}{s} \left\| \sum_{i \in [s]} S_i((\mathbf{A} + \mathbf{G}_i)\mathbf{x}_{\text{OPT}} - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 + \lambda \cdot \|\mathbf{x}_{\text{OPT}}\|^2 \end{aligned}$$

which likewise implies

$$\|\mathbf{x}'\| \leq \frac{1}{\lambda} \cdot \left(\frac{1}{s} \left\| \sum_{i \in [s]} S_i((\mathbf{A} + \mathbf{G}_i)\mathbf{x}_{\text{OPT}} - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 + \lambda \cdot \|\mathbf{x}_{\text{OPT}}\|^2 \right).$$

Insertion both bounds back into Equation 7, we obtain

$$\begin{aligned} & \|\mathbf{Ax}' - \mathbf{b}\|^2 + \lambda \|\mathbf{x}'\|^2 \\ & \leq (1 + \alpha_S)^2 \\ & \left(\frac{1}{s} \left\| \sum_{i \in [s]} S_i((\mathbf{A} + \mathbf{G}_i)\mathbf{x}_{\text{OPT}} - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 + \lambda \|\mathbf{x}_{\text{OPT}}\|^2 \right) \\ & + 2 \left(1 + \frac{1}{\alpha_S} \right) \eta \sigma^2 n d \log \frac{1}{\beta} + 2 \left(1 + \frac{1}{\alpha_S} \right) \frac{\eta \sigma^2 n d \log \frac{1}{\beta}}{\lambda} \quad (8) \\ & \cdot \left(\frac{1}{s} \left\| \sum_{i \in [s]} S_i((\mathbf{A} + \mathbf{G}_i)\mathbf{x}_{\text{OPT}} - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 + \lambda \cdot \|\mathbf{x}_{\text{OPT}}\|^2 \right) \quad (9) \end{aligned}$$

We now turn our attention to $\left\| \sum_{i \in [s]} S_i((\mathbf{A} + \mathbf{G}_i)\mathbf{x}_{\text{OPT}} - (\mathbf{b} + \mathbf{g}_i)) \right\|^2$. Using Lemma 5.5 and Equation 5, we have

$$\begin{aligned} & \frac{1}{s} \left\| \sum_{i \in [s]} S_i((\mathbf{A} + \mathbf{G}_i)\mathbf{x}_{\text{OPT}} - (\mathbf{b} + \mathbf{g}_i)) \right\|^2 + \lambda \cdot \|\mathbf{x}_{\text{OPT}}\|^2 \\ & \leq (1 + \alpha_S) \frac{1}{s} \left\| \sum_{i \in [s]} S_i(\mathbf{Ax}_{\text{OPT}} - \mathbf{b}) \right\|^2 \\ & + \left(1 + \frac{1}{\alpha_S} \right) \frac{1}{s} \left\| \sum_{i \in [s]} S_i(\mathbf{G}_i\mathbf{x}_{\text{OPT}} - \mathbf{g}_i) \right\|^2 + \lambda \cdot \|\mathbf{x}_{\text{OPT}}\|^2 \\ & \leq (1 + \alpha_S)^2 \|\mathbf{Ax}_{\text{OPT}} - \mathbf{b}\|^2 \\ & + \left(1 + \frac{1}{\alpha_S} \right) \eta \sigma^2 n d \log \frac{1}{\beta} (\|\mathbf{x}_{\text{OPT}}\|^2 + 1) + \lambda \|\mathbf{x}_{\text{OPT}}\|^2 \\ & \leq (1 + \alpha_S)^2 \left(\|\mathbf{Ax}_{\text{OPT}} - \mathbf{b}\|^2 + \lambda \|\mathbf{x}_{\text{OPT}}\|^2 \right) \\ & + \left(1 + \frac{1}{\alpha_S} \right) \frac{\eta \sigma^2 n d \log \frac{1}{\beta}}{\lambda} (\lambda \|\mathbf{x}_{\text{OPT}}\|^2 + \lambda) \\ & \leq \left((1 + \alpha_S)^2 + \left(1 + \frac{1}{\alpha_S} \right) \frac{\eta \sigma^2 n d \log \frac{1}{\beta}}{\lambda} \right) \\ & \left(\|\mathbf{Ax}_{\text{OPT}} - \mathbf{b}\|^2 + \lambda \cdot \|\mathbf{x}_{\text{OPT}}\|^2 \right) \\ & + \left(1 + \frac{1}{\alpha_S} \right) \eta \sigma^2 n d \log \frac{1}{\beta} \end{aligned}$$

Inserting this into Equation 9 and collecting all the terms, we obtain

$$\begin{aligned} & \|\mathbf{Ax}' - \mathbf{b}\|^2 + \lambda \|\mathbf{x}'\|^2 \\ & \leq \left(\frac{32}{\alpha_S} \left(\frac{\eta \sigma^2 n d \log \frac{1}{\beta}}{\lambda} + \frac{2}{\alpha_S} \left(\frac{\eta \sigma^2 n d \log \frac{1}{\beta}}{\lambda} \right)^2 \right) \right) \quad (10) \\ & + (1 + \alpha_S)^4 \left(\|\mathbf{Ax}_{\text{OPT}} - \mathbf{b}\|^2 + \lambda \|\mathbf{x}_{\text{OPT}}\|^2 \right) \\ & + \frac{32}{\alpha_S} \left(\frac{\eta \sigma^2 n d \log \frac{1}{\beta}}{\lambda} + \frac{2}{\alpha_S} \left(\frac{\eta \sigma^2 n d \log \frac{1}{\beta}}{\lambda} \right)^2 \right) \end{aligned}$$

By our choice of σ^2 from Theorem 4.7 and using the sketching matrix of [22] (see Table 2), we have $\sigma^2 \cdot n \in O(d^2 \log^4 d \cdot \alpha_S^{-4} \cdot \log^4 \frac{1}{\beta} \varepsilon^{-2} \cdot \log \frac{1}{\delta})$. Thus, we have

$$\begin{aligned} & \|\mathbf{Ax}' - \mathbf{b}\|^2 + \lambda \|\mathbf{x}'\|^2 \leq (\|\mathbf{Ax}_{\text{OPT}} - \mathbf{b}\|^2 + \lambda \|\mathbf{x}_{\text{OPT}}\|^2) \cdot \left(1 + 15\alpha_S \right. \\ & \left. + \tilde{O} \left(\frac{d^3 \alpha_S^{-5} \log^5 \frac{1}{\beta} \cdot \varepsilon^{-2} \log \frac{1}{\delta}}{\lambda} + \frac{d^6 \alpha_S^{-10} \log^{10} \frac{1}{\beta} \cdot \varepsilon^{-4} \log^2 \frac{1}{\delta}}{\lambda^2} \right) \right) \\ & \left. + \tilde{O} \left(\frac{d^3 \alpha_S^{-5} \log^5 \frac{1}{\beta} \cdot \varepsilon^{-2} \log \frac{1}{\delta}}{\lambda} + \frac{d^6 \alpha_S^{-10} \log^{10} \frac{1}{\beta} \cdot \varepsilon^{-4} \log^2 \frac{1}{\delta}}{\lambda^2} \right) \right) \quad \square \end{aligned}$$

Note that we can get a constant multiplicative error as long as the regularization factor λ depends on a sufficiently large polynomial in d , while being independent of n . Generally such a relationship still has good generalization properties when training a regression model. If λ is sufficiently large, one should choose α_S to be as small as possible, minimizing the tradeoff between α_S and $\frac{\alpha_S^5}{\lambda}$. If λ is not sufficiently large, one should choose $\alpha_S = 1/3$ such that \mathbf{S} is an oblivious subspace embedding, but the target dimension does not have a prohibitively large dependency on the sketch distortion.

6 Experimental Evaluation

Section 6.1 provides the running time experiments and in Section 6.2 we discuss our utility investigation both for ridge regression and low-rank approximation.

6.1 Running Time of LTM

The only overhead of the LTM over the local model is the execution of the linear transform in MPC, and comes from distributing $\mathbf{A} + \mathbf{G}_i \in \mathbb{R}^{n \times d}$ among the servers for $i \in [s]$ and then performing matrix multiplications $S_i(\mathbf{A} + \mathbf{G}_i)$ in MPC, where S_1, \dots, S_s is a decomposition of some $\mathbf{S} \in \mathbb{R}^{m \times n}$ chosen according to [22, 58] with sparsity s . We implemented our approach in MP-SPDZ [47], a popular and easy to use MPC framework, for multiple combinations of n and s , using S servers from Amazon Web Services t3.large instances. We use additive secret sharing over the ring of integers modulo 2^{64} , which results in no communication between servers, and semi-honest security against an adversary that corrupts all but one servers.

We fix the dimensionality $d = 10$ and $m = 50$, and vary the number of clients $n \in \{100000, 250000, 500000, 750000, 1000000\}$, the sparsity $s \in \{1, 10, 20, 30, 40, 50\}$ of the sketch (which dictates how many linear transformations we need to apply) and the number of servers $S \in \{2, 3\}$. The data matrix \mathbf{A} is generated at random such that all entries are smaller than 2^{32} and the decomposition $\mathbf{S}_1, \dots, \mathbf{S}_s$ of some sketch \mathbf{S} is generated according to our mechanism. Every parameter setting is then evaluated by running the protocol 10 times and averaging over the running times. Table 3 shows the runtime for each server and the total communication load for one particular combination of parameters, which includes three servers: Even with one million clients, the computation on each server lasts less than 2 seconds. We conclude that the use of MPC in the LTM does not hinder the computational efficiency of our proposed mechanisms, showing that they can easily be used in practice. Table 4 reports communication and time required when running the protocol with two servers, and Table 5 reports time and communication requirements when varying s .

Table 3: Computation cost T_{MPC} and communication cost C_{MPC} of the LTM using MPC for varying number of clients n . Here $m = 100$, $d = 10$, $s = 1$ and we are working with $S = 3$ servers.

n	T_{MPC} (sec)	C_{MPC} (MB)
100000	0.172 ± 0.019	24.024
250000	0.457 ± 0.058	60.024
500000	0.861 ± 0.102	120.024
750000	1.219 ± 0.128	180.024
1000000	1.641 ± 0.165	240.024

Table 4: Computation cost T_{MPC} and communication cost C_{MPC} of the LTM using MPC for varying number of clients n . Here $m = 100$, $d = 10$, $s = 1$ and we are working with $S = 2$ servers.

n	T_{MPC} (sec)	C_{MPC} (MB)
100000	0.177 ± 0.028	16.016
250000	0.453 ± 0.036	40.016
500000	0.824 ± 0.101	80.016
750000	1.240 ± 0.157	120.016
1000000	1.683 ± 0.130	160.016

6.2 Interpolation between Local and Central DP

In contrast to the central model of differential privacy, the performance of differentially private outputs in the local model decreases with increasing number of clients n . For regression and matrix factorization problems, our theoretical results for the LTM and the Gaussian mechanism suggest that this performance decrease can be avoided without sacrificing privacy under reasonable assumptions. It is natural to ask for which parametrizations of the Gaussian mechanism the performance remains acceptable. If more noise still yields good performance, we can tolerate more corrupted clients.

Table 5: Computation cost T_{MPC} and communication cost C_{MPC} of the LTM using MPC for varying sparsity s . Here $m = 100$, $d = 10$, $n = 500000$ and we are working with $S = 3$ servers.

s	T_{MPC} (sec)	C_{MPC} (MB)
1	0.851 ± 0.099	80.016
10	6.863 ± 0.388	800.016
20	13.002 ± 0.445	1600.016
30	20.356 ± 1.171	2400.016
40	28.023 ± 0.660	3200.016
50	33.708 ± 1.384	4000.016

Table 6: Parameters of real-world datasets.

Dataset	n	d	k
Power [41]	2049280	6	3
Elevation [46]	434874	2	1
Ethylene [32]	4178504	18	5
Songs [11]	515345	89	15

Therefore, we evaluate empirically how the error of ridge regression and low-rank approximation develops as n grows in practice and against synthetic benchmarks.

The experiments described here thus aim to demonstrate that, as the number of clients increases, the asymptotic error of differential privacy mechanisms in the LTM is between the error in the central and local models.

Setup. Our Gaussian mechanism in the LTM adds noise sampled from $\mathcal{N}(0, \sigma^2)$ to every entry in input data matrix \mathbf{A} . The variance σ^2 depends on privacy parameters ϵ and δ and is chosen proportional to n^{-p} for $p \in [0, 1]$. This enables us to interpolate between the local model ($p = 0$) and the LTM ($p = 1$). For each dataset we measure error for the central model, the Laplace mechanism and the Gaussian mechanism with $p \in \{1, 0.9, 0.8, 0.7, 0.6, 0.5, 0\}$, by reporting the average error over 20 runs of the algorithms per dataset. The Laplace mechanism in the LTM on the other hand adds the sum of m samples from $\text{Lap}(0, 1/\epsilon)$ to every entry in \mathbf{SA} in order to investigate the $\delta = 0$ regime in the LTM.

As a baseline for low-rank approximation in the central model, we implemented MOD-SULQ [17], an approach where $\mathbf{A}^T \mathbf{A}$ is perturbed by adding Gaussian noise. We measure the error in terms of the excess risk;

$$\psi = \frac{\|\mathbf{A} - \mathbf{A}\mathbf{X}'\mathbf{X}'^T\|_F^2 - \|\mathbf{A} - \mathbf{A}\mathbf{X}_{\text{OPT}}\mathbf{X}_{\text{OPT}}^T\|_F^2}{n},$$

where \mathbf{X}_{OPT} denotes the optimal solution and \mathbf{X}' denotes the solution after adding noise to the training data. We vary the privacy parameter k based on the dataset at hand.

For linear regression in the central model, we implemented the so-called Sufficient Statistics Perturbation (SSP) algorithm [70]. Due to its singular dependency on a multiplicative error, we use approximation factor as an error measure:

$$\phi = \frac{\|\mathbf{A}\mathbf{x}' - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}'\|_2^2}{\|\mathbf{A}\mathbf{x}_{\text{OPT}} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}_{\text{OPT}}\|_2^2}$$

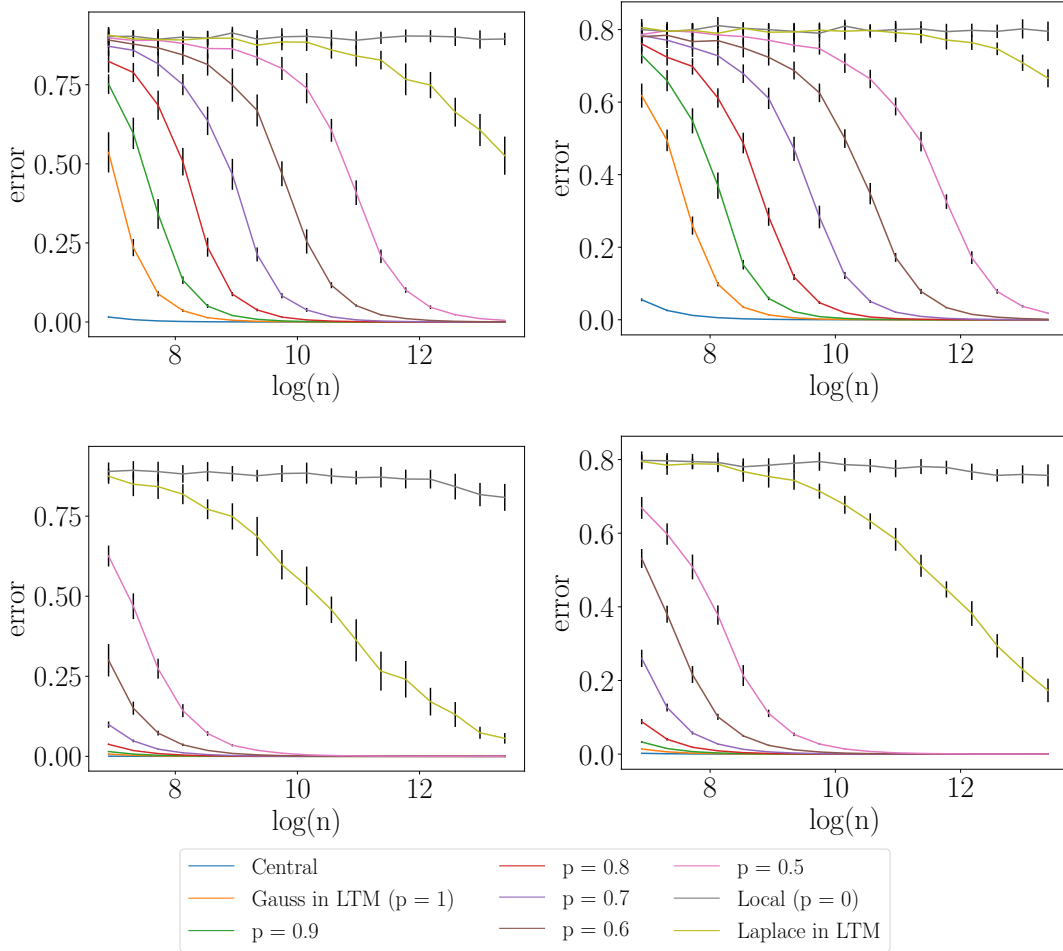


Figure 2: Plots depicting the asymptotic behavior of error ψ for $\epsilon \in \{0.1, 0.5\}$ (top, bottom) and $k \in \{5, 10\}$ (left, right), with $d = 50$. The gray line depicts the error of the local mechanism, and the orange and lime lines depict our approach using Gaussian and Laplacian noise respectively. The other lines resemble different values of p when using Gaussian noise. The standard deviations are depicted by the vertical black lines and the x -axis is logarithmic in the number of clients n .

Table 7: Experimental evaluation of error ψ on real-world datasets, including standard deviations. Here we are in the setting where $\epsilon = 0.05$.

Dataset	Local	LTM Laplace	LTM Gauss	Central
Power	$(1.206 \pm 0.651) \times 10^{-2}$	$(4.036 \pm 2.448) \times 10^{-3}$	$(2.108 \pm 2.400) \times 10^{-6}$	$(1.763 \pm 0.621) \times 10^{-6}$
Elevation	$(1.193 \pm 0.896) \times 10^{-1}$	$(1.520 \pm 4.484) \times 10^{-2}$	$(4.718 \pm 4.201) \times 10^{-8}$	$(4.465 \pm 4.836) \times 10^{-8}$
Ethylene	$(5.877 \pm 1.256) \times 10^{-4}$	$(5.821 \pm 1.074) \times 10^{-4}$	$(2.420 \pm 0.486) \times 10^{-6}$	$(7.901 \pm 1.909) \times 10^{-6}$
Songs	$(5.617 \pm 0.283) \times 10^{-7}$	$(5.629 \pm 0.311) \times 10^{-7}$	$(5.556 \pm 0.347) \times 10^{-7}$	$(5.613 \pm 0.324) \times 10^{-7}$

where \mathbf{x}_{OPT} denotes the optimal solution and \mathbf{x}' denotes the solution after adding noise to the training data.

Datasets. We evaluate our mechanism on synthetic and real datasets, enabling us to vary the number of sampled points n and interpolating between datasets of different sizes.

The synthetic datasets for low-rank approximation have a large spectral gap from the k th to the $(k+1)$ st singular value with random

bases, which emphasizes the performance difference between the various private mechanisms. We generate synthetic data by first producing a matrix \mathbf{A}' where every entry is sampled from $\mathcal{N}(0, 1)$ and then changing its singular values such that there are exactly k big ones and the rest are small. More specifically, we set $\mathbf{A} = \mathbf{U}'\Sigma\mathbf{V}'$, where $\mathbf{U}'\Sigma\mathbf{V}'$ is the SVD of \mathbf{A} and Σ is a diagonal matrix with the

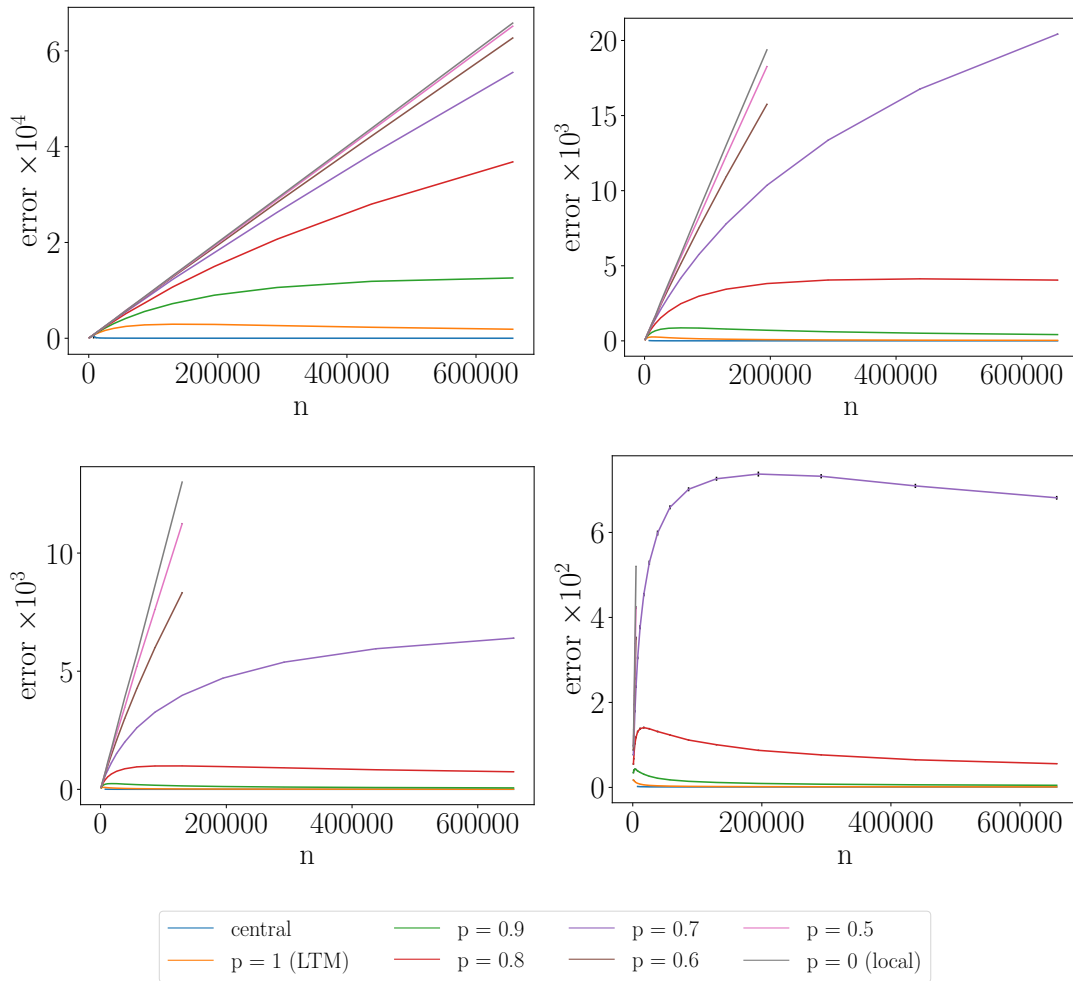


Figure 3: Plots depicting the asymptotic behavior of error ϕ for $\epsilon \in \{0.01, 0.03, 0.05, 0.1\}$ (top left, top right, bottom left, bottom right), with $d = 10$, $\lambda = 10$ and $\mu^2 = n$. The grey line depicts the error of the local mechanism, the blue line does it in the central model and the orange one depicts our approach. The other lines resemble different values of p . In most cases the standard deviations are so small, that it is not possible to see those.

first k values set to $\sqrt{n/k}$ and the rest set to $1/n$. We vary parameter $k \in \{5, 10\}$ for $d = 50$.

We generate synthetic data for regression by first sampling every entry in \mathbf{A} from $\mathcal{N}(0, 1)$, then sampling \mathbf{x}_{OPT} such that all entries are sampled from $\mathcal{N}(0, \mu^2)$, and finally setting $\mathbf{b} = \mathbf{A}\mathbf{x}_{\text{OPT}}$. We vary parameters $d \in \{3, 10, 50\}$, and $\mu^2 \in \{1, n, n^2\}$.

For all combinations of parameters, we generate 17 synthetic datasets of sizes $\{1000^{i-1.5} | i = 0, \dots, 16\}$. For each of those we then measure error for the central mechanism, the Laplace mechanism and the Gaussian mechanism with $p \in \{1, 0.9, 0.8, 0.7, 0.6, 0.5, 0\}$, by running the algorithms 20 times per dataset and reporting the average error.

In addition to synthetic datasets, we also evaluated our mechanism on 4 datasets from the UC Irvine Machine Learning Repository. Table 6 provides their number of entries n and dimensionality d ,

as well as the rank k we chose for low-rank approximation experiments.

- The first dataset consists of electric power consumption measurements in one household [41] and the feature we try to predict is `sub_metering_3`. We ignore the date and time features and the data points that had missing values. This leaves us with 6 features (plus the one we are predicting) and 2049280 data points.
- The Elevation dataset [46] consists of 434874 open street map elevation measurements from North Jutland, Denmark. We predict the elevation from the longitude and latitude features.
- The Ethylene dataset [32] contains recordings of sensors exposed to a mixture of gas. We trained on the part where the sensors were exposed to a mixtures of Ethylene and CO

Table 8: Experimental evaluation of error ϕ on real-world datasets, including standard deviations. Here we are in the setting where $\epsilon = 0.03$ and $\lambda = 10$.

Dataset	Local	Our	Central
Power	2.364 \pm .007	1.055 \pm .000	1.001 \pm .001
Elevation	1.939 \pm .008	1.000 \pm .000	1.000 \pm .000
Ethylene	3.125 \pm .002	1.000 \pm .000	1.000 \pm .000
Songs	20.64 \pm .016	1.000 \pm .000	1.000 \pm .000

in air. The feature we are predicting is the last one TGS2620, which leaves us with $d = 18$ and $n = 4178504$.

- The Songs dataset [11], consists of 89 audio features that are meant to predict the release year of a song.

Low-Rank Approximation. Figure 2 shows our results for two privacy regimes ($\epsilon \in \{0.1, 0.5\}$) with synthetic data and Table 7 shows the error for $\epsilon = 0.05$ with real-world datasets. In all chosen parameter settings, we observe that as n grows, the error in the LTM asymptotically approaches the error in the central model, both on real and synthetic data sets. On real-world datasets, our Gaussian approach performs significantly better than the Gaussian mechanism in the local model and is very close to the central MOD-SULQ [17] mechanism. Both with synthetic and real-world datasets, our Laplace approach performs better than the local model, though not as significantly as the Gaussian approach. However, this is the price of achieving pure ϵ -differential privacy with $\delta = 0$.

Ridge Regression. Figure 3 provides our experimental results based on synthetic data, and Table 8 shows the error resulting from real-world datasets. As expected, our approach performs asymptotically better than the Gaussian mechanism in the local model, but worse than SSP in the central model. We also found that as n increases, the error of our approach asymptotically approaches the error in the central model. For $p = 0.9, 0.8, 0.7, 0.6$, the error eventually decreases for a sufficiently large n for the privacy regimes we considered ($\epsilon \in \{0.01, 0.03, 0.05, 0.1\}$).

For $p = 0.5$ though we saw a significant jump towards the local model in terms of asymptotic behaviour. In all the settings we considered, $p = 0.5$ showed an asymptotic increase of the error. We tested on synthetic datasets of up to $n = 40$ million, and the error also increases in this regime.

Table 8 shows the results we got by applying the local, our mechanism and SSP on real-world data. They include standard deviations, though for many of the results those are so small that they appear as 0 in the table. We found, that our mechanism performs significantly better than the Gaussian mechanism for local privacy. In most it even performs so well that there is basically no difference to the central model.

Varying λ . In Figure 4, we provide plots for the same setting of parameters as on Figure 3 ($d = 10$ and $\mu^2 = n$). Though, we also vary $\lambda \in \{1, 10, 100\}$ here. The experiments show that varying λ does not change the asymptotic behavior of any of the mechanisms we investigated, in the sense that a variance proportional to $n^{-0.6}$ or lower eventually had decreasing error, while a variance proportional to $n^{0.5}$ produced increasing errors. The increase of λ does

however produce a significantly lower error in all settings (note the y -axes on the figure). Our theoretical results likewise predict this behaviour, as while the error bounds of Theorem 5.8 improve with increasing λ , they do not affect the dependency on n . Thus, we view the experiments as a confirmation that the theoretical bounds, while potentially improvable, express the correct asymptotic relationship between parameters and approximation bounds.

Acknowledgments

The research described in this paper has received funding from: the European Research Council (ERC) under the European Unions’s Horizon 2020 research and innovation programme under grant agreement No 803096 (SPEC); the Danish Independent Research Council under Grant-ID DFF-2064-00016B (YOSO); DFF-3103-00077B (CryptoDigi) and the Danish Independent Research Council under Grant-ID 1051-001068 (ACBD).

References

- [1] Dimitris Achlioptas. 2003. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. *J. Comput. Syst. Sci.* 66, 4 (2003), 671–687. [https://doi.org/10.1016/S0022-0000\(03\)00025-4](https://doi.org/10.1016/S0022-0000(03)00025-4)
- [2] Apple and Google. 2021. *Exposure Notification Privacy-preserving Analytics (ENPA)*. White Paper.
- [3] Raman Arora, Vladimir Braverman, and Jalaj Upadhyay. 2018. Differentially Private Robust Low-Rank Approximation. In *Advances in Neural Information Processing Systems*.
- [4] Sanjeev Arora, Elad Hazan, and Satyen Kale. 2006. A Fast Random Sampling Algorithm for Sparsifying Matrices. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 9th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2006 and 10th International Workshop on Randomization and Computation, RANDOM 2006, Barcelona, Spain, August 28-30 2006, Proceedings (Lecture Notes in Computer Science, Vol. 4110)*, Josep Díaz, Klaus Jansen, José D. P. Rolim, and Uri Zwick (Eds.). Springer, 272–279. https://doi.org/10.1007/11830924_26
- [5] Maria-Florina Balcan, Simon Shaolei Du, Yining Wang, and Adams Wei Yu. 2016. An Improved Gap-Dependency Analysis of the Noisy Power Method. In *29th Annual Conference on Learning Theory (Proceedings of Machine Learning Research, Vol. 49)*, Vitaly Feldman, Alexander Rakhlin, and Ohad Shamir (Eds.). PMLR, Columbia University, New York, New York, USA, 284–309. <https://proceedings.mlr.press/v49/balcan16a.html>
- [6] Raef Bassily, Kobbi Nissim, Uri Stemmer, and Abhradeep Thakurta. 2020. Practical Locally Private Heavy Hitters. *J. Mach. Learn. Res.* 21 (2020), 16:1–16:42. <https://jmlr.org/papers/v21/18-786.html>
- [7] Raef Bassily and Adam D. Smith. 2015. 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*, Rocco A. Servedio and Ronitt Rubinfeld (Eds.). ACM, 127–135. <https://doi.org/10.1145/2746539.2746632>
- [8] Raef Bassily, Adam D. Smith, and Abhradeep Thakurta. 2014. Private Empirical Risk Minimization: Efficient Algorithms and Tight Error Bounds. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS*.
- [9] Luca Becchetti, Marc Bury, Vincent Cohen-Addad, Fabrizio Grandoni, and Chris Schwiegelshohn. 2019. Oblivious dimension reduction for k -means: beyond subspaces and the Johnson-Lindenstrauss lemma. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC*.
- [10] Amos Beimel, Iftach Haitner, Kobbi Nissim, and Uri Stemmer. 2020. On the Round Complexity of the Shuffle Model. In *Theory of Cryptography: 18th International Conference, TCC 2020, Durham, NC, USA, November 16–19, 2020, Proceedings, Part II*. Springer-Verlag, Berlin, Heidelberg, 683–712. https://doi.org/10.1007/978-3-030-64378-2_24
- [11] T. Bertin-Mahieux. 2011. YearPredictionMSD. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C50K61>
- [12] Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnes, and Bernhard Seefeld. 2017. Prochlo: Strong Privacy for Analytics in the Crowd. In *Proceedings of the 26th Symposium on Operating Systems Principles (SOSP)*.
- [13] Jeremiah Blocki, Avrim Blum, Anupam Datta, and Or Sheffet. 2012. The Johnson-Lindenstrauss Transform Itself Preserves Differential Privacy. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*. IEEE Computer Society, 410–419. <https://doi.org/10.1109/FOCS.2012.67>

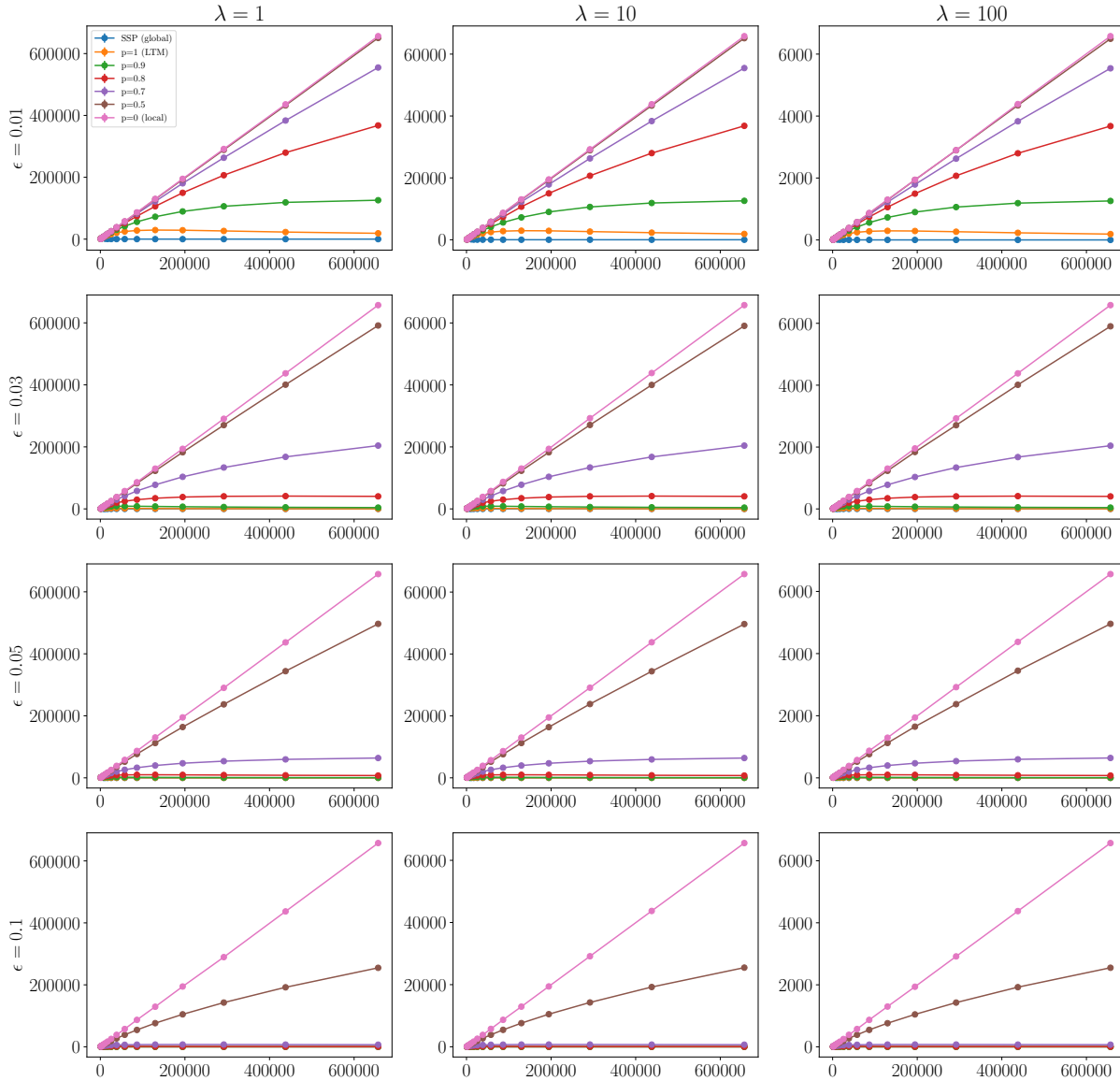


Figure 4: Plots depicting the error ϕ as n increases, for $\epsilon \in \{0.01, 0.03, 0.05, 0.1\}$ (top to bottom) and $\lambda \in \{1, 10, 100\}$ (left to right). The middle row depicts the same choice of parameters as Figure 3.

[14] Avrim Blum, Cynthia Dwork, Frank McSherry, and Kobbi Nissim. 2005. Practical privacy: the SuLQ framework. In *Proceedings of the Twenty-Fourth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems* (, Baltimore, Maryland.) (PODS '05). Association for Computing Machinery, New York, NY, USA, 128–138. <https://doi.org/10.1145/1065167.1065184>

[15] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical Secure Aggregation for Privacy-Preserving Machine Learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS)*.

[16] T. Tony Cai, Yichen Wang, and Linjun Zhang. 2021. The cost of privacy: Optimal rates of convergence for parameter estimation with differential privacy. *The Annals of Statistics* 49, 5 (2021), 2825 – 2850. <https://doi.org/10.1214/21-AOS2058>

[17] Kamalika Chaudhuri, Anand D. Sarwate, and Kaushik Sinha. 2012. Near-optimal Differentially Private Principal Components. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger (Eds.), 998–1006. <https://proceedings.neurips.cc/paper/2012/hash/f770b62bc8f42a0b66751fe636fc6eb0-Abstract.html>

[18] Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. 2019. Distributed Differential Privacy via Shuffling. In *Advances in Cryptology EUROCRYPT*.

[19] Albert Cheu and Chao Yan. 2023. Necessary Conditions in Multi-Server Differential Privacy. In *14th Innovations in Theoretical Computer Science Conference (ITCS)*.

[20] Kenneth L. Clarkson and David P. Woodruff. 2009. Numerical linear algebra in the streaming model. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, Michael Mitzenmacher (Ed.), ACM, 205–214. <https://doi.org/10.1145/1536414.1536445>

[21] Kenneth L. Clarkson and David P. Woodruff. 2013. Low rank approximation and regression in input sparsity time. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, Dan Boneh, Tim Roughgarden, and Joan Feigenbaum (Eds.), ACM, 81–90. <https://doi.org/10.1145/>

- 2488608.2488620
- [22] Michael B. Cohen. 2016. Nearly Tight Oblivious Subspace Embeddings by Trace Inequalities. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, Robert Krauthgamer (Ed.). SIAM, 278–287. <https://doi.org/10.1137/1.9781611974331.CH21>
- [23] Michael B. Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu. 2015. Dimensionality Reduction for k-Means Clustering and Low Rank Approximation. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, Rocco A. Servedio and Ronitt Rubinfeld (Eds.). ACM, 163–172. <https://doi.org/10.1145/2746539.2746569>
- [24] Ronald Cramer, Ivan Bjerre Damgård, and Jesper Buus Nielsen. 2015. *Secure Multiparty Computation and Secret Sharing*. Cambridge University Press.
- [25] Ivan Damgård, Hannah Keller, Boel Nelson, Claudio Orlandi, and Rasmus Pagh. 2023. Differentially Private Selection from Secure Distributed Computing. In *The Web Conference (WWW)*.
- [26] Christos Dimitrakakis, Blaine Nelson, Aikaterini Mitrokoatsa, and Benjamin I. P. Rubinfeld. 2014. Robust and Private Bayesian Inference. In *Algorithmic Learning Theory*. Springer International Publishing, Cham, 291–305.
- [27] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In *Advances in Cryptology - EUROCRYPT*.
- [28] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of Cryptography*.
- [29] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends in Theoretical Computer Science* (2014).
- [30] Cynthia Dwork, Kunal Talwar, Abhradeep Thakurta, and Li Zhang. 2014. Analyze gauss: optimal bounds for privacy-preserving principal component analysis. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, David B. Shmoys (Ed.). ACM, 11–20. <https://doi.org/10.1145/2591796.2591883>
- [31] Úlfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Abhradeep Thakurta. 2019. Amplification by Shuffling: From Local to Central Differential Privacy via Anonymity. In *Proceedings of the 2019 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*.
- [32] Jordi Fonollosa. 2015. Gas sensor array under dynamic gas mixtures. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5WP4C>.
- [33] James Foulds, Joseph Geumleke, Max Welling, and Kamalika Chaudhuri. 2016. On the Theory and practice of privacy-preserving Bayesian data analysis. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence (Jersey City, New Jersey, USA) (UAI'16)*. AUAI Press, Arlington, Virginia, USA, 192–201.
- [34] Badih Ghazi, Noah Golowich, Ravi Kumar, Rasmus Pagh, and Ameya Velingker. 2021. On the Power of Multiple Anonymous Messages: Frequency Estimation and Selection in the Shuffle Model of Differential Privacy. In *Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part III (Lecture Notes in Computer Science, Vol. 12698)*, Anne Canteaut and François-Xavier Standaert (Eds.). Springer, 463–488. https://doi.org/10.1007/978-3-030-77883-5_16
- [35] Badih Ghazi, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, and Adam Sealfon. 2023. On Computing Pairwise Statistics with Local Differential Privacy. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (Eds.). http://papers.nips.cc/paper_files/paper/2023/hash/5642b9811a9ac5281be1cc84c275f251-Abstract-Conference.html
- [36] Badih Ghazi, Ravi Kumar, Pasin Manurangsi, and Rasmus Pagh. 2020. Private Counting from Anonymous Messages: Near-Optimal Accuracy with Vanishing Communication Overhead. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 3505–3514. <http://proceedings.mlr.press/v119/ghazi20a.html>
- [37] Sławomir Gorczycka and Li Xiong. 2017. A Comprehensive Comparison of Multi-party Secure Additions with Differential Privacy. *IEEE Transactions on Dependable and Secure Computing* (2017).
- [38] Moritz Hardt and Eric Price. 2014. The Noisy Power Method: A Meta Algorithm with Applications. In *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger (Eds.), Vol. 27. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2014/file/729c68884bd359ade15d5f163166738a-Paper.pdf
- [39] Moritz Hardt and Aaron Roth. 2012. Beating randomized response on incoherent matrices. In *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing (New York, New York, USA) (STOC '12)*. Association for Computing Machinery, New York, NY, USA, 1255–1268. <https://doi.org/10.1145/2213977.2214088>
- [40] Moritz Hardt and Aaron Roth. 2013. Beyond worst-case analysis in private singular vector computation. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing (Palo Alto, California, USA) (STOC '13)*. Association for Computing Machinery, New York, NY, USA, 331–340. <https://doi.org/10.1145/2488608.2488650>
- [41] Georges Hebrail and Alice Berard. 2012. Individual household electric power consumption. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C58K54>.
- [42] Mikael Møller Høgsgaard, Lior Kamma, Kasper Green Larsen, Jelani Nelson, and Chris Schwiegelshohn. 2023. Sparse Dimensionality Reduction Revisited. *CoRR abs/2302.06165* (2023). <https://doi.org/10.48550/ARXIV.2302.06165> arXiv:2302.06165
- [43] Daniel M. Kane and Jelani Nelson. 2012. Sparser Johnson-Lindenstrauss transforms. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, Yuval Rabani (Ed.). SIAM, 1195–1206. <https://doi.org/10.1137/1.9781611973099.94>
- [44] Michael Kapralov and Kunal Talwar. [n.d.]. *On differentially private low rank approximation*. 1395–1414. <https://doi.org/10.1137/1.9781611973105.101> arXiv:https://epubs.siam.org/doi/pdf/10.1137/1.9781611973105.101
- [45] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2008. What Can We Learn Privately?. In *49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*.
- [46] Manohar Kaul. 2013. 3D Road Network (North Jutland, Denmark). UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5GP51>.
- [47] Marcel Keller. 2020. MP-SPDZ: A Versatile Framework for Multi-Party Computation. In *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna (Eds.). ACM, 1575–1590. <https://doi.org/10.1145/3372297.3417872>
- [48] Krishnaram Kenthapadi, Aleksandra Korolova, Ilya Mironov, and Nina Mishra. 2013. Privacy via the Johnson-Lindenstrauss Transform. *J. Priv. Confidentiality* 5, 1 (2013). <https://doi.org/10.29012/JPC.V5I1.625>
- [49] Daniel Kifer, Adam D. Smith, and Abhradeep Thakurta. 2012. Private Convex Optimization for Empirical Risk Minimization with Applications to High-dimensional Regression. In *The 25th Annual Conference on Learning Theory (COLT)*.
- [50] Beatrice Laurent and Pascal Massart. 2000. Adaptive estimation of a quadratic functional by model selection. *Annals of statistics* (2000), 1302–1338.
- [51] Yingyu Liang, Maria-Florina Balcan, Vandana Kanchanapally, and David P. Woodruff. 2014. Improved Distributed Principal Component Analysis. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger (Eds.). 3113–3121. <https://proceedings.neurips.cc/paper/2014/hash/52947e0ade57a09e4a1386d08f17b656-Abstract.html>
- [52] Xiyang Liu, Weihao Kong, and Seewong Oh. 2022. Differential privacy and robust statistics in high dimensions. In *Conference on Learning Theory*.
- [53] Luca Melis, George Danezis, and Emiliano De Cristofaro. 2016. Efficient Private Statistics with Succinct Sketches. <https://doi.org/10.14722/ndss.2016.23175>
- [54] Xiangrui Meng and Michael W. Mahoney. 2013. Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, Dan Boneh, Tim Roughgarden, and Joan Feigenbaum (Eds.). ACM, 91–100. <https://doi.org/10.1145/2488608.2488621>
- [55] Jason Milionis, Alkis Kalavasis, Dimitris Fotakis, and Stratis Ioannidis. 2022. Differentially Private Regression with Unbounded Covariates. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 151)*, Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (Eds.). PMLR, 3242–3273. <https://proceedings.mlr.press/v151/milionis22a.html>
- [56] Kentaro Minami, Hltomi Arai, Issei Sato, and Hiroshi Nakagawa. 2016. Differential Privacy without Sensitivity. In *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Eds.), Vol. 29. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2016/file/a7aeed74714116f3b292a982238f83d2-Paper.pdf
- [57] Vaikkunth Mugunthan, Antigoni Polychroniadou, David Byrd, and Tucker Hybinette Balch. 2019. Smpai: Secure multi-party computation for federated learning. In *Proceedings of the NeurIPS 2019 Workshop on Robust AI in Financial Services*.
- [58] Jelani Nelson and Huy L. Nguyen. 2013. OSNAP: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS*.
- [59] Aleksandar Nikolov. 2023. Private Query Release via the Johnson-Lindenstrauss Transform. In *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, Nikhil Bansal and Viswanath Nagarajan (Eds.). SIAM, 4982–5002. <https://doi.org/10.1137/1.9781611977554.CH182>
- [60] Rasmus Pagh and Mikkel Thorup. 2022. Improved Utility Analysis of Private CountSketch. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran

- Associates, Inc., 25631–25643. https://proceedings.neurips.cc/paper_files/paper/2022/file/a47f5cdf1469751597d78e803fc590f-Paper-Conference.pdf
- [61] Gilles Pisier. 1999. *The volume of convex bodies and Banach space geometry*. Cambridge Tracts in Mathematics. 94.
- [62] Tamás Sarlós. 2006. Improved Approximation Algorithms for Large Matrices via Random Projections. In *47th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2006)*, 21–24 October 2006, Berkeley, California, USA, *Proceedings*. IEEE Computer Society, 143–152. <https://doi.org/10.1109/FOCS.2006.37>
- [63] Or Sheffet. 2019. Old Techniques in Differentially Private Linear Regression. In *Proceedings of the 30th International Conference on Algorithmic Learning Theory (Proceedings of Machine Learning Research, Vol. 98)*, Aurélien Garivier and Satyen Kale (Eds.). PMLR, 789–827. <https://proceedings.mlr.press/v98/sheffet19a.html>
- [64] Nina Mesing Stausholm. 2021. Improved Differentially Private Euclidean Distance Approximation. In *PODS’21: Proceedings of the 40th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, Virtual Event, China, June 20–25, 2021*, Leonid Libkin, Reinhard Pichler, and Paolo Guagliardo (Eds.). ACM, 42–56. <https://doi.org/10.1145/3452021.3458328>
- [65] Nina Mesing Stausholm. 2021. Improved Differentially Private Euclidean Distance Approximation. In *PODS’21: Proceedings of the 40th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, Virtual Event, China, June 20–25, 2021*, Leonid Libkin, Reinhard Pichler, and Paolo Guagliardo (Eds.). ACM, 42–56. <https://doi.org/10.1145/3452021.3458328>
- [66] Thomas Steinke. 2020. Multi-Central Differential Privacy. <https://doi.org/10.48550/ARXIV.2009.05401>
- [67] Kunal Talwar, Shan Wang, Audra McMillan, Vojta Jina, Vitaly Feldman, Bailey Basile, Aine Cahill, Yi Sheng Chan, Mike Chatzidakis, Junye Chen, Oliver Chick, Mona Chitnis, Suman Ganta, Yusuf Goren, Filip Granqvist, Kristine Guo, Frederic Jacobs, Omid Javidbakht, Albert Liu, Richard Low, Dan Mascenik, Steve Myers, David Park, Wonhee Park, Gianni Parsa, Tommy Pauly, Christian Priebe, Rehan Rishi, Guy Rothblum, Michael Scaria, Linmao Song, Congzheng Song, Karl Tarbe, Sebastian Vogt, Luke Winstrom, and Shundong Zhou. 2023. Samplable Anonymous Aggregation for Private Federated Data Analysis. arXiv:2307.15017 [cs.CR] <https://arxiv.org/abs/2307.15017>
- [68] Jalaj Upadhyay. 2018. The Price of Privacy for Low-rank Factorization. In *Advances in Neural Information Processing Systems*.
- [69] Prateek Varshney, Abhradeep Thakurta, and Prateek Jain. 2022. (Nearly) Optimal Private Linear Regression for Sub-Gaussian Data via Adaptive Clipping. In *Proceedings of Thirty Fifth Conference on Learning Theory (Proceedings of Machine Learning Research, Vol. 178)*, Po-Ling Loh and Maxim Raginsky (Eds.). PMLR, 1126–1166. <https://proceedings.mlr.press/v178/varshney22a.html>
- [70] Duy Vu and Aleksandra B. Slavkovic. 2009. Differential Privacy for Clinical Trial Data: Preliminary Evaluations. In *ICDM Workshops 2009, IEEE International Conference on Data Mining*.
- [71] Di Wang, Marco Gaboardi, and Jinhui Xu. 2018. Empirical Risk Minimization in Non-interactive Local Differential Privacy Revisited. In *Advances in Neural Information Processing Systems*.
- [72] Di Wang, Lijie Hu, Huanyu Zhang, Marco Gaboardi, and Jinhui Xu. 2023. Generalized Linear Models in Non-interactive Local Differential Privacy with Public Data. *Journal of Machine Learning Research* 24, 132 (2023), 1–57. <http://jmlr.org/papers/v24/21-0523.html>
- [73] Di Wang, Adam D. Smith, and Jinhui Xu. 2018. Noninteractive Locally Private Learning of Linear Models via Polynomial Approximations. In *International Conference on Algorithmic Learning Theory*. <https://api.semanticscholar.org/CorpusID:67856419>
- [74] Yu-Xiang Wang. 2018. Revisiting differentially private linear regression: optimal and adaptive prediction & estimation in unbounded domain. In *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI*.
- [75] Yu-Xiang Wang, Stephen E. Fienberg, and Alexander J. Smola. 2015. Privacy for Free: Posterior Sampling and Stochastic Gradient Monte Carlo. In *Proceedings of the 32nd International Conference on Machine Learning, ICML*.
- [76] Stanley L. Warner. 1965. Randomized Response: A Survey Technique for Eliminating Evasive Answer Bias. *J. Amer. Statist. Assoc.* (1965).
- [77] David P. Woodruff. 2014. Sketching as a Tool for Numerical Linear Algebra. *Found. Trends Theor. Comput. Sci.* 10, 1–2 (2014), 1–157. <https://doi.org/10.1561/0400000060>
- [78] Fuheng Zhao, Dan Qiao, Rachel Redberg, Divyakant Agrawal, Amr El Abbadi, and Yu-Xiang Wang. 2022. Differentially private linear sketches: efficient implementations and applications. In *Proceedings of the 36th International Conference on Neural Information Processing Systems (New Orleans, LA, USA) (NIPS ’22)*. Curran Associates Inc., Red Hook, NY, USA, Article 922, 14 pages.
- [79] Kai Zheng, Wenlong Mou, and Liwei Wang. 2017. Collect at once, use effectively: making non-interactive locally private learning possible. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*.