Differentially Private Ordinary Least Squares

Or Sheffet [1]

Abstract

Linear regression is one of the most prevalent techniques in machine learning; however, it is also common to use linear regression for its *explanatory* capabilities rather than label prediction. Ordinary Least Squares (OLS) is often used in statistics to establish a correlation between an attribute (e.g. gender) and a label (e.g. income) in the presence of other (potentially correlated) features. OLS assumes a particular model that randomly generates the data, and derives *t-values* — representing the likelihood of each real value to be the true correlation. Using t-values, OLS can release a *confidence interval*, which is an interval on the reals that is likely to contain the true correlation; and when this interval does not intersect the origin, we can *reject the null hypothesis* as it is likely that the true correlation is non-zero. Our work aims at achieving similar guarantees on data under differentially private estimators. First, we show that for well-spread data, the Gaussian Johnson-Lindenstrauss Transform (JLT) gives a very good approximation of t-values; secondly, when JLT approximates Ridge regression (linear regression with l_2-regularization) we derive, under certain conditions, confidence intervals using the projected data; lastly, we derive, under different conditions, confidence intervals for the "Analyze Gauss" algorithm (Dwork et al., 2014).

1. Introduction

Since the early days of differential privacy, its main goal was to design privacy preserving versions of existing techniques for data analysis. It is therefore no surprise that several of the first differentially private algorithms were machine learning algorithms, with a special emphasis on the ubiquitous problem of linear regression (Kasiviswanathan et al., 2008; Chaudhuri et al., 2011; Kifer et al., 2012; Bassily et al., 2014). However, *all* existing body of work on differentially private linear regression measures utility by bounding the distance between the linear regressor found by the standard non-private algorithm and the regressor found by the privacy-preserving algorithm. This is motivated from a machine-learning perspective, since bounds on the difference in the estimators translate to error bounds on prediction (or on the loss function). Such bounds are (highly) interesting and non-trivial, yet they are of little use in situations where one uses linear regression to establish correlations rather than predict labels.

In the statistics literature, Ordinary Least Squares (OLS) is a technique that uses linear regression in order to infer the correlation between a variable and an outcome, especially in the presence of other factors. And so, in this paper, we draw a distinction between "linear regression," by which we refer to the machine learning technique of finding a specific estimator for a specific loss function; and "Ordinary Least Squares," by which we refer to the statistical inference done assuming a specific model for generating the data and that uses linear regression. Many argue that OLS is the most prevalent technique in social sciences (Agresti & Finlay, 2009). Such works make no claim as to the labels of a new unlabeled batch of samples. Rather they aim to establish the existence of a strong correlation between the label and some feature. Needless to say, in such works, the privacy of individuals' data is a concern.

In order to determine that a certain variable x_j is positively (resp. negatively) correlated with an outcome y, OLS assumes a model where the outcome y is a noisy version of a linear mapping of all variables: $y = \boldsymbol{\beta} \cdot \boldsymbol{x} + e$ (with e denoting random Gaussian noise) for some predetermined and unknown $\boldsymbol{\beta}$. Then, given many samples $(\boldsymbol{x}_i, y_i)$ OLS establishes two things: (i) when fitting a linear function to best predict y from $\boldsymbol{x}$ *over the sample* (via computing $\hat{\boldsymbol{\beta}} = \left(\sum_i \boldsymbol{x}_i \boldsymbol{x}_i^\mathsf{T}\right)^{-1} \left(\sum_i y_i \boldsymbol{x}_i\right)$) the coefficient $\hat{\beta}_j$ is positive (resp. negative); and (ii) *inferring*, based on $\hat{\beta}_j$, that the true β_j is likely to reside in $\mathbb{R}_{>0}$ (resp. $\mathbb{R}_{<0}$). In fact, the crux in OLS is by describing β_j using a probability distribution over the reals, indicating where β_j is likely to fall, derived by computing *t-values*. These values take into account both the variance in the data as well as the variance of the noise e.[1] Based on this probability distribution one can

[1] Computing Science Dept., University of Alberta, Edmonton AB, Canada. This work was done when the author was at Harvard University, supported by NSF grant CNS-123723. Correspondence to: Or Sheffet <osheffet@ualberta.ca>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017. Copyright 2017 by the author(s).

[1] For example, imagine we run linear regression on a certain

define the α*-confidence interval* — an interval I centered at $\hat{\beta}_j$ whose likelihood to contain β_j is $1-\alpha$. Of particular importance is the notion of *rejecting the null-hypothesis*, where the interval I does not contain the origin, and so one is able to say with high confidence that β_j is positive (resp. negative). Further details regarding OLS appear in Section 2.

In this work we give the *first* analysis of statistical inference for OLS using differentially private estimators. We emphasize that the novelty of our work does not lie in the differentially-private algorithms, which are, as we discuss next, based on the Johnson-Lindenstrauss Transform (JLT) and on additive Gaussian noise and are already known to be differentially private (Blocki et al., 2012; Dwork et al., 2014). Instead, the novelty of our work lies in the analyses of the algorithms and in proving that the output of the algorithms is useful for statistical inference.

The Algorithms. Our first algorithm (Algorithm 1) is an adaptation of Gaussian JLT. Proving that this adaptation remains (ϵ, δ)-differentially private is straightforward (the proof appears in Appendix A.1). As described, the algorithm takes as input a parameter r (in addition to the other parameters of the problem) that indicates the number of rows in the JL-matrix. Later, we analyze what should one set as the value of r. Our second algorithm is taken verbatim from the work of Dwork et al (2014). We deliberately focus on algorithms that approximate the 2^{nd}-moment matrix of the data and then run hypothesis-testing by post-processing the output, for two reasons. First, they enable sharing of data[2] and running unboundedly many hypothesis-tests. Since, we do not deal with OLS based on the private single-regression ERM algorithms (Chaudhuri et al., 2011; Bassily et al., 2014) as such inference requires us to use the Fisher-information matrix of the loss function — but these algorithms do not minimize a private loss-function but rather prove that outputting the minimizer of the perturbed loss-function is private. This means that differentially-private OLS based on these ERM algorithms requires us to devise new versions of these algorithms, making this a second step in this line of work... (After first understanding what we can do using existing algorithms.) We leave this approach — as well as performing private hypothesis testing using a PTR-type algorithm (Dwork & Lei, 2009) (output merely reject / don't-reject decision without justification), or releasing only relevant tests judging by their p-values (Dwork et al., 2015) — for future work.

Algorithm 1 Outputting a private Johnson-Lindenstrauss projection of a matrix.

```
Input: A matrix $A \in \mathbb{R}^{n \times d}$ and a bound $B > 0$ on the $l_2$-norm of any row in $A$.
Privacy parameters: $\epsilon, \delta > 0$.
Parameter $r$ indicating the number of rows in the resulting matrix.
Set $w$ s.t. $w^2 = \frac{8B^2}{\epsilon}\left(\sqrt{2r\ln(8/\delta)} + 2\ln(8/\delta)\right)$.
Sample $Z \sim Lap(4B^2/\epsilon)$ and let $\sigma_{\min}(A)$ denote the smallest singular value of $A$.
if $\sigma_{\min}(A)^2 > w^2 + Z + \frac{4B^2\ln(1/\delta)}{\epsilon}$ then
    Sample a $(r \times n)$-matrix $R$ whose entries are i.i.d samples from a normal Gaussian.
    return $RA$ and "matrix unaltered".
else
    Let $A'$ denote the result of appending $A$ with the $d \times d$-matrix $wI_{d \times d}$.
    Sample a $(r \times (n+d))$-matrix $R$ whose entries are i.i.d samples from a normal Gaussian.
    return$RA'$ and "matrix altered".
end if
```

Algorithm 2 "Analyze Gauss" Algorithm of Dwork et al (2014).

```
Input: A matrix $A \in \mathbb{R}^{n \times d}$ and a bound $B > 0$ on the $l_2$-norm of any row in $A$.
Privacy parameters: $\epsilon, \delta > 0$.
$N \leftarrow$ symmetric $(d \times d)$-matrix with upper triangle entries sampled i.i.d from $\mathcal{N}\left(0, \frac{2B^4\ln(2/\delta)}{\epsilon^2}\right)$.
return $A^{\mathsf{T}}A + N$.
```

Our Contribution and Organization. We analyze the performances of our algorithms on a matrix A of the form $A = [X; \boldsymbol{y}]$, where each coordinate y_i is generated according to the *homoscedastic model* with Gaussian noise, which is a classical model in statistics. We assume the existence of a vector $\boldsymbol{\beta}$ s.t. for every i we have $y_i = \boldsymbol{\beta}^{\mathsf{T}}\boldsymbol{x}_i + e_i$ and e_i is sampled i.i.d from $\mathcal{N}(0, \sigma^2)$.[3]

We study the result of running Algorithm 1 on such data in the two cases: where A wasn't altered by the algorithm and when A was appended by the algorithm. In the former case, Algorithm 1 boils down to projecting the data under a Gaussian JLT. Sarlos (2006) has already shown that the JLT is useful for linear regression, yet his work bounds the l_2-norm of the difference between the estimated re-

$(X, \boldsymbol{y})$ which results in a vector $\hat{\boldsymbol{\beta}}$ with coordinates $\hat{\beta}_1 = \hat{\beta}_2 = 0.1$. Yet while the column X_1 contains many 1s and (-1)s, the column X_2 is mostly populated with zeros. In such a setting, OLS gives that it is likely to have $\beta_1 \approx 0.1$, whereas no such guarantees can be given for β_2.

[2] Researcher A collects the data and uses the approximation of the 2^{nd}-moment matrix to test some OLS hypothesis; but once the approximation is published researcher B can use it to test for a completely different hypothesis.

[3] This model may seem objectionable. Assumptions like the noise independence, 0-meaned or sampled from a Gaussian distribution have all been called into question in the past. Yet due to the prevalence of this model we see fit to initiate the line of work on differentially private Least Squares with this Ordinary model.

gression before and after the projection. Following Sarlos' work, other works in statistics have analyzed compressed linear regression (Zhou et al., 2007; Pilanci & Wainwright, 2014a;b). However, none of these works give confidence intervals based on the projected data, presumably for three reasons. Firstly, these works are motivated by computational speedups, and so they use fast JLT as opposed to our analysis which leverages on the fact that our JL-matrix is composed of i.i.d Gaussians. Secondly, the focus of these works is not on OLS but rather on newer versions of linear regression, such as Lasso or when $\boldsymbol{\beta}$ lies in some convex set. Lastly, it is evident that the smallest confidence interval is derived from the data itself. Since these works do not consider privacy applications, (actually, (Zhou et al., 2007; Pilanci & Wainwright, 2014a) do consider privacy applications of the JLT, but quite different than differential privacy) they assume the analyst has access to the data itself, and so there was no need to give confidence intervals for the projected data. Our analysis is therefore the first, to the best of our knowledge, to derive t-values — and therefore achieve all of the rich expressivity one infers from t-values, such as confidence bounds and null-hypotheses rejection — for OLS estimations *without having access to X itself*. We also show that, under certain conditions, the sample complexity for correctly rejecting the null-hypothesis increases from a certain bound N_0 (without privacy) to a bound of $N_0 + \tilde{O}(\sqrt{N_0} \cdot \kappa(\frac{1}{n}A^\mathsf{T}A)/\epsilon)$ with privacy (where $\kappa(M)$ denotes the condition number of the matrix M.) This appears in Section 3.

In Section 4 we analyze the case Algorithm 1 does append the data and the JLT is applied to A'. In this case, solving the linear regression problem on the projected A' approximates the solution for *Ridge Regression* (Tikhonov, 1963; Hoerl & Kennard, 1970). In Ridge Regression we aim to solve $\min_{\boldsymbol{z}} \left(\sum_i (y_i - \boldsymbol{z}^\mathsf{T}\boldsymbol{x}_i)^2 + w^2\|\boldsymbol{z}\|^2\right)$, which means we penalize vectors whose l_2-norm is large. In general, it is not known how to derive t-values from Ridge regression, and the literature on deriving confidence intervals solely from Ridge regression is virtually non-existent. Indeed, prior to our work there was no need for such calculations, as access to the data was (in general) freely given, and so deriving confidence intervals could be done by appealing back to OLS. We too are unable to derive approximated t-values in the general case, but under additional assumptions about the data — which admittedly depend in part on $\|\boldsymbol{\beta}\|$ and so cannot be verified solely from the data — we show that solving the linear regression problem on RA' allows us to give confidence intervals for β_j, thus correctly determining the correlation's sign.

In Section 5 we discuss the "Analyze Gauss" algorithm (Dwork et al., 2014) that outputs a noisy version of a covariance of a given matrix using additive noise rather than multiplicative noise. Empirical work (Xi et al., 2011) shows that Analyze Gauss's output might be non-PSD if the input has small singular values, and this results in truly bad regressors. Nonetheless, under additional conditions (that imply that the output is PSD), we derive confidence bounds for Dwork et al's "Analyze Gauss" algorithm. Finally, in Section 6 we experiment with the heuristic of computing the t-values directly from the outputs of Algorithms 1 and 2. We show that Algorithm 1 is more "conservative" than Algorithm 2 in the sense that it tends to not reject the null-hypothesis until the number of examples is large enough to give a very strong indication of rejection. In contrast, Algorithm 2 may wrongly rejects the null-hypothesis even when it is true.

Discussion. Some works have already looked at the intersection of differentially privacy and statistics (Dwork & Lei, 2009; Smith, 2011; Chaudhuri & Hsu, 2012; Duchi et al., 2013; Dwork et al., 2015) (especially focusing on robust statistics and rate of convergence). But only a handful of works studied the significance and power of hypotheses testing under differential privacy, without arguing that the noise introduced by differential privacy vanishes asymptotically (Vu & Slavkovic, 2009; Uhler et al., 2013; Wang et al., 2015; Rogers et al., 2016). These works are experimentally promising, yet they (i) focus on different statistical tests (mostly Goodness-of-Fit and Independence testing), (ii) are only able to prove results for the case of simple hypothesis-testing (a single hypothesis) with an efficient data-generation procedure through repeated simulations — a cumbersome and time consuming approach. In contrast, we deal with a composite hypothesis (we simultaneously reject all $\boldsymbol{\beta}$s with $sign(\beta_j) \neq sign(\hat{\beta}_j)$) by altering the confidence interval (or the critical region).

One potential reason for avoiding confidence-interval analysis for differentially private hypotheses testing is that it does involve re-visiting existing results. Typically, in statistical inference the sole source of randomness lies in the underlying model of data generation, whereas the estimators themselves are a deterministic function of the dataset. In contrast, differentially private estimators are inherently random *in their computation*. Statistical inference that considers *both* the randomness in the data and the randomness in the computation is highly uncommon, and this work, to the best of our knowledge, is the first to deal with randomness in OLS hypothesis testing. We therefore strive in our analysis to separate the two sources of randomness — as in classic hypothesis testing, we use α to denote the bound on any bad event that depends solely on the homoscedastic model, and use ν to bound any bad event that depends on the randomized algorithm.[4] (Thus, any result which is originally of the form "α-reject the null-hypothesis" is now converted into a result "$(\alpha+\nu)$-reject the null hypothesis".)

[4] Or any randomness in generating the feature matrix X which standard OLS theory assumes to be fixed, see Theorems 2.2 and 3.3.

2. Preliminaries and OLS Background

Notation. Throughout this paper, we use $lower$-case letters to denote scalars (e.g., y_i or e_i); ***bold*** characters to denote vectors; and UPPER-case letters to denote matrices. The l-dimensional all zero vector is denoted $\mathbf{0}_l$, and the $l \times m$-matrix of all zeros is denoted $0_{l\times m}$. We use $\boldsymbol{e}$ to denote the specific vector $\boldsymbol{y} - X\boldsymbol{\beta}$ in our model; and though the reader may find it a bit confusing but hopefully clear from the context — we also use $\boldsymbol{e}_j$ and $\boldsymbol{e}_k$ to denote elements of the natural basis (unit length vector in the direction of coordinate j or k). We use ϵ, δ to denote the privacy parameters of Algorithms 1 and 2, and use α and ν to denote confidence parameters (referring to bad events that hold w.p. $\leq \alpha$ and $\leq \nu$ resp.) based on the homoscedastic model or the randomized algorithm resp. We also stick to the notation from Algorithm 1 and use w to denote the positive scalar for which $w^2 = \frac{8B^2}{\epsilon}\left(\sqrt{2r\ln(8/\delta)} + \ln(8/\delta)\right)$ throughout this paper. We use standard notation for SVD composition of a matrix ($M = U\Sigma V^{\mathsf{T}}$), its singular values and its Moore-Penrose inverse (M^+).

The Gaussian distribution. A univariate Gaussian $\mathcal{N}(\mu, \sigma^2)$ denotes the Gaussian distribution whose mean is μ and variance σ^2. Standard concentration bounds on Gaussians give that $\mathbf{Pr}[x > \mu + 2\sigma\sqrt{\ln(2/\nu)}] < \nu$ for any $\nu \in (0, \frac{1}{e})$. A multivariate Gaussian $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ for some positive semi-definite Σ denotes the multivariate Gaussian distribution where the mean of the j-th coordinate is the μ_j and the covariance between coordinates j and k is $\Sigma_{j,k}$. The PDF of such Gaussian is defined only on the subspace $colspan(\Sigma)$. A matrix Gaussian distribution, denoted $\mathcal{N}(M_{a\times b}, I_{a\times a}, V)$ has mean M, independence among its rows and variance V for each of its columns. We also require the following property of Gaussian random variables: Let X and Y be two random Gaussians s.t. $X \sim \mathcal{N}(0, \sigma^2)$ and $Y \sim \mathcal{N}(0, \lambda^2)$ where $1 \leq \frac{\sigma^2}{\lambda^2} \leq c^2$ for some c, then for any $S \subset \mathbb{R}$ we have $\frac{1}{c}\mathbf{Pr}_{x\leftarrow Y}[x \in S] \leq \mathbf{Pr}_{x\leftarrow X}[x \in S] \leq c\mathbf{Pr}_{x\leftarrow Y}[x \in S/c]$ (see Proposition A.2).

Additional Distributions. We denote by $Lap(\sigma)$ the *Laplace distribution* whose mean is 0 and variance is $2\sigma^2$. The χ^2_k*-distribution*, where k is referred to as the degrees of freedom of the distribution, is the distribution over the l_2-norm squared of the sum of k independent normal Gaussians. That is, given i.i.d $X_1, \ldots, X_k \sim \mathcal{N}(0,1)$ it holds that $\boldsymbol{\zeta} \overset{\text{def}}{=} (X_1, X_2, \ldots, X_k) \sim \mathcal{N}(\mathbf{0}_k, I_{k\times k})$, and $\|\boldsymbol{\zeta}\|^2 \sim \chi^2_k$. Existing tail bounds on the χ^2_k distribution (Laurent & Massart, 2000) give that $\mathbf{Pr}\left[\|\boldsymbol{\zeta}\|^2 \in (\sqrt{k} \pm \sqrt{2\ln(2/\nu)})^2\right] \geq 1 - \nu$. The T_k-distribution, where k is referred to as the degrees of freedom of the distribution, denotes the distribution over the reals created by *independently* sampling $Z \sim \mathcal{N}(0,1)$ and $\|\zeta\|^2 \sim \chi^2_k$, and taking the quantity $Z/\sqrt{\|\zeta\|^2/k}$. It is a known fact that $T_k \overset{k\to\infty}{\to} \mathcal{N}(0,1)$, thus it is a common practice to apply Gaussian tail bounds to the T_k-distribution when k is sufficiently large.

Differential Privacy. In this work, we deal with input in the form of a $n \times d$-matrix with each row bounded by a l_2-norm of B. Two inputs A and A' are called *neighbors* if they differ on a single row.

Definition 2.1 ((Dwork et al., 2006a)). *An algorithm* ALG *which maps* $(n \times d)$*-matrices into some range* $\mathcal{R}$ *is* (ϵ, δ)- differential privacy *it holds that* $\mathbf{Pr}[\mathsf{ALG}(A) \in \mathcal{S}] \leq e^\epsilon \mathbf{Pr}[\mathsf{ALG}(A') \in \mathcal{S}] + \delta$ *for all neighboring inputs* A *and* A' *and all subsets* $\mathcal{S} \subset \mathcal{R}$.

Background on OLS. For the unfamiliar reader, we give here a *very* brief overview of the main points in OLS. Further details, explanations and proofs appear in Section A.3.

We are given n observations $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ where $\forall i, \boldsymbol{x}_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$. We assume the existence of $\boldsymbol{\beta} \in \mathbb{R}^p$ s.t. the label y_i was derived by $y_i = \boldsymbol{\beta}^{\mathsf{T}}\boldsymbol{x}_i + e_i$ where $e_i \sim \mathcal{N}(0, \sigma^2)$ independently (also known as the homoscedastic Gaussian model). We use the matrix notation where X denotes the $(n \times p)$- feature matrix and $\boldsymbol{y}$ denotes the labels. We assume X has full rank.

The parameters of the model are therefore $\boldsymbol{\beta}$ and σ^2, which we set to discover. To that end, we minimize $\min_{\boldsymbol{z}} \|\boldsymbol{y} - X\boldsymbol{z}\|^2$ and have

$$\hat{\boldsymbol{\beta}} = (X^{\mathsf{T}}X)^{-1}X^{\mathsf{T}}\boldsymbol{y} = (X^{\mathsf{T}}X)^{-1}X^{\mathsf{T}}(X\boldsymbol{\beta} + \boldsymbol{e}) = \boldsymbol{\beta} + X^+\boldsymbol{e} \quad (1)$$

$$\boldsymbol{\zeta} = \boldsymbol{y} - X\hat{\boldsymbol{\beta}} = (X\boldsymbol{\beta} + \boldsymbol{e}) - X(\boldsymbol{\beta} + X^+\boldsymbol{e}) = (I - XX^+)\boldsymbol{e} \quad (2)$$

And then for any coordinate j the *t-value*, which is the quantity $t(\beta_j) \overset{\text{def}}{=} \frac{\hat{\beta}_j - \beta_j}{\sqrt{(X^{\mathsf{T}}X)^{-1}_{j,j} \cdot \frac{\|\boldsymbol{\zeta}\|}{\sqrt{n-p}}}}$, is distributed according to T_{n-p}-distribution. I.e., $\mathbf{Pr}\left[\hat{\boldsymbol{\beta}} \text{ and } \boldsymbol{\zeta} \text{ satisfying } t(\beta_j) \in S\right] = \int_S \mathsf{PDF}_{T_{n-p}}(x)dx$ for any measurable $S \subset \mathbb{R}$. Thus $t(\beta_j)$ describes the likelihood of any β_j — for any $z \in \mathbb{R}$ we can now give an estimation of how likely it is to have $\beta_j = z$ (which is $\mathsf{PDF}_{T_{n-p}}(t(z))$), and this is known as *t-test* for the value z. In particular, given $0 < \alpha < 1$, we denote c_α as the number for which the interval $(-c_\alpha, c_\alpha)$ contains a probability mass of $1 - \alpha$ from the T_{n-p}-distribution. And so we derive a corresponding *confidence interval* I_α centered at $\hat{\beta}_j$ where $\beta_j \in I_\alpha$ with confidence of level of $1 - \alpha$.

Of particular importance is the quantity $t_0 \overset{\text{def}}{=} t(0) = \frac{\hat{\beta}_j\sqrt{n-p}}{\|\boldsymbol{\zeta}\|\sqrt{(X^{\mathsf{T}}X)^{-1}_{j,j}}}$,since if there is no correlation between x_j and y then the likelihood of seeing $\hat{\beta}_j$ depends on the ratio of its magnitude to its standard deviation. As mentioned earlier, since $T_k \overset{k\to\infty}{\to} \mathcal{N}(0,1)$, then rather than viewing this t_0 as sampled from a T_{n-p}-distribution, it is common to think of t_0 as a sample from a normal Gaussian $\mathcal{N}(0,1)$. This allows us to associate t_0 with a p-value, estimating the event "β_j and $\hat{\beta}_j$ have different signs." Specifically, given $\alpha \in (0, 1/2)$, we *α-reject the null hypothesis* if $p_0 < \alpha$. Let τ_α be the number s.t. $\Phi(\tau_\alpha) = \int_{\tau_\alpha}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx = \alpha$. This means we α-reject the null hypothesis when $|t_0| > \tau_\alpha$. We now lower bound the number of i.i.d sample points needed in order to α-reject the null hypothesis. This bound is our basis for comparison between standard OLS and the differentially private version.[5]

Theorem 2.2. *Fix any positive definite matrix $\Sigma \in \mathbb{R}^{p\times p}$ and any $\nu \in (0, \frac{1}{2})$. Fix parameters $\boldsymbol{\beta} \in \mathbb{R}^p$ and σ^2 and a coordinate j s.t. $\beta_j \neq 0$. Let X be a matrix whose n rows are i.i.d samples from $\mathcal{N}(\mathbf{0}, \Sigma)$, and $\boldsymbol{y}$ be a vector where $y_i - (X\boldsymbol{\beta})_i$ is sampled i.i.d from $\mathcal{N}(0, \sigma^2)$. Fix $\alpha \in (0,1)$. Then w.p. $\geq 1 - \alpha - \nu$ we have that OLS's $(1-\alpha)$-confidence interval has length $O(c_\alpha \sqrt{\sigma^2/(n\sigma_{\min}(\Sigma))})$ provided $n \geq C_1(p + \ln(1/\nu))$ for some sufficiently large constant C_1. Furthermore, there exists a constant C_2 such that w.p. $\geq 1 - \alpha - \nu$ OLS (correctly) rejects the null hypothesis provided $n \geq \max\left\{ C_1(p + \ln(1/\nu)),\ p + C_2 \frac{\sigma^2}{\beta_j^2} \cdot \frac{c_\alpha^2 + \tau_\alpha^2}{\sigma_{\min}(\Sigma)} \right\}$, where c_α is the number for which $\int_{-c_\alpha}^{c_\alpha} \mathsf{PDF}_{T_{n-p}}(x)dx = 1 - \alpha$.*

3. OLS over Projected Data

In this section we deal with the output of Algorithm 1 in the special case where Algorithm 1 outputs `matrix unaltered` and so we work with RA.

To clarify, the setting is as follows. We denote $A = [X; \boldsymbol{y}]$ the column-wise concatenation of the $(n \times (d-1))$-matrix X with the n-length vector $\boldsymbol{y}$. (Clearly, we can denote any column of A as $\boldsymbol{y}$ and any subset of the remaining columns as the matrix X.) We therefore denote the output $RA = [RX; R\boldsymbol{y}]$ and for simplicity we denote $M = RX$ and $p = d - 1$. We denote the SVD decomposition of $X = U\Sigma V^{\mathsf{T}}$. So U is an orthonormal basis for the column-span of X and as X is full-rank V is an orthonormal basis for $\mathbb{R}^p$. Finally, in our work we examine the linear regression problem derived from the projected data. That is, we denote

$$\tilde{\boldsymbol{\beta}} = (X^{\mathsf{T}} R^{\mathsf{T}} RX)^{-1} (RX)^{\mathsf{T}} (R\boldsymbol{y}) = \boldsymbol{\beta} + (RX)^{+} R\boldsymbol{e} \quad (3)$$

$$\tilde{\sigma}^2 = \frac{r}{r-p} \|\tilde{\boldsymbol{\zeta}}\|^2 \text{ , with } \tilde{\boldsymbol{\zeta}} = \tfrac{1}{\sqrt{r}} R\boldsymbol{y} - \tfrac{1}{\sqrt{r}} (RX)\tilde{\boldsymbol{\beta}} \quad (4)$$

We now give our main theorem, for estimating the t-values based on $\tilde{\boldsymbol{\beta}}$ and $\tilde{\sigma}$.

Theorem 3.1. *Let X be a $(n \times p)$-matrix, and parameters $\boldsymbol{\beta} \in \mathbb{R}^p$ and σ^2 are such that we generate the vector $\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{e}$ with each coordinate of $\boldsymbol{e}$ sampled independently from $\mathcal{N}(0, \sigma^2)$. Assume Algorithm 1 projects the matrix $A = [X; \boldsymbol{y}]$ without altering it. Fix $\nu \in (0, 1/2)$ and $r = p + \Omega(\ln(1/\nu))$. Fix coordinate j. Then we have that w.p. $\geq 1 - \nu$ deriving $\tilde{\boldsymbol{\beta}}$ and $\tilde{\sigma}^2$ as in Equations (3) and (4), the pivot quantity $\tilde{t}(\beta_j) = \frac{\tilde{\beta}_j - \beta_j}{\tilde{\sigma}\sqrt{(X^{\mathsf{T}} R^{\mathsf{T}} RX)^{-1}_{j,j}}}$ has a distribution $\mathcal{D}$ satisfying $e^{-a}\mathsf{PDF}_{T_{r-p}}(x) \leq \mathsf{PDF}_{\mathcal{D}}(x) \leq e^{a}\mathsf{PDF}_{T_{r-p}}(e^{-a}x)$ for any $x \in \mathbb{R}$, where we denote $a = \frac{r-p}{n-p}$.*

The implications of Theorem 3.1 are immediate: all estimations one can do based on the t-values from the true data $X, \boldsymbol{y}$, we can now do based on $\tilde{t}$ modulo an approximation factor of $\exp(\frac{r-p}{n-p})$. In particular, Theorem 3.1 enables us to deduce a corresponding confidence interval based on $\tilde{\boldsymbol{\beta}}$.

Corollary 3.2. *In the same setting as in Theorem 3.1, w.p. $\geq 1 - \nu$ we have the following. Fix any $\alpha \in (0, \frac{1}{2})$. Let $\tilde{c}_\alpha$ denote the number s.t. the interval $(\tilde{c}_\alpha, \infty)$ contains $\frac{\alpha}{2}e^{-a}$ probability mass of the T_{r-p}-distribution. Then $\mathbf{Pr}[\beta_j \in \left(\tilde{\beta}_j \pm e^a \cdot \tilde{c}_\alpha \cdot \tilde{\sigma}\sqrt{(X^{\mathsf{T}} R^{\mathsf{T}} RX)^{-1}_{j,j}}\right)] \geq 1 - \alpha$.* [6]

We compare the confidence interval of Corollary 3.2 to the confidence interval of the standard OLS model, whose length is $c_\alpha \frac{\|\boldsymbol{\zeta}\|}{\sqrt{n-p}} \sqrt{(X^{\mathsf{T}} X)^{-1}_{j,j}}$. As R is a JL-matrix, known results regarding the JL transform give that $\|\tilde{\boldsymbol{\zeta}}\| = \Theta(\|\boldsymbol{\zeta}\|)$, and that $\sqrt{(r-p)(X^{\mathsf{T}} R^{\mathsf{T}} RX)^{-1}_{j,j}} = \Theta\left(\sqrt{(X^{\mathsf{T}} X)^{-1}_{j,j}}\right)$. We therefore have that $\tilde{\sigma}\sqrt{(X^{\mathsf{T}} R^{\mathsf{T}} RX)^{-1}_{j,j}} = \frac{\|\tilde{\boldsymbol{\zeta}}\|\sqrt{r}}{\sqrt{r-p}} \sqrt{(X^{\mathsf{T}} R^{\mathsf{T}} RX)^{-1}_{j,j}} = \sqrt{\frac{r\cdot(n-p)}{(r-p)^2}} \cdot \Theta\left(\frac{\|\boldsymbol{\zeta}\|}{\sqrt{n-p}} \sqrt{(X^{\mathsf{T}} X)^{-1}_{j,j}}\right)$. So for values of r for which $\frac{r}{r-p} = \Theta(1)$ we get that the confidence interval of Theorem 3.1 is a factor of $\Theta\left(\frac{\tilde{c}_\alpha}{c_\alpha}\sqrt{\frac{n-p}{r-p}}\right)$-larger than the standard OLS confidence interval. Observe that when $\alpha = \Theta(1)$, which is the common case, the dominating factor is $\sqrt{(n-p)/(r-p)}$. This bound intuitively makes sense: we have contracted n observations to r observations, hence our model is based on confidence intervals derived from T_{r-p} rather than T_{n-p}.

In the supplementary material we give further discussion, in which we compare our work to the more straight-forward bounds one gets by "plugging in" Sarlos' work (2006); and we also compare ourselves to the bounds derived from alternative works in differentially private linear regression.

[5] Theorem 2.2 also illustrates how we "separate" the two sources of privacy. In this case, ν bounds the probability of bad events that depend to sampling the rows of X, and α bounds the probability of a bad event that depends on the sampling of the $\boldsymbol{y}$ coordinates.

[6] Moreover, this interval is essentially optimal: denote $\tilde{d}_\alpha$ s.t the interval $(\tilde{d}_\alpha, \infty)$ contains $\frac{\alpha}{2}e^{\frac{r-p}{n-p}}$ probability mass of the T_{r-p}-distribution. Then $\mathbf{Pr}[\beta_j \in \left(\tilde{\beta}_j \pm \tilde{d}_\alpha \cdot \tilde{\sigma}\sqrt{(X^{\mathsf{T}} R^{\mathsf{T}} RX)^{-1}_{j,j}}\right)] \leq 1 - \alpha$.

Rejecting the Null Hypothesis. Due to Theorem 3.1, we can mimic OLS' technique for rejecting the null hypothesis. I.e., we denote $\tilde{t}_0 = \frac{\tilde{\beta}_j}{\tilde{\sigma}\sqrt{(X^\mathsf{T}R^\mathsf{T}RX)^{-1}_{j,j}}}$ and reject the null-hypothesis if indeed the associated $\tilde{p}_0$, denoting p-value of the slightly truncated $e^{-\frac{r-p}{n-p}}\tilde{t}_0$, is below $\alpha \cdot e^{-\frac{r-p}{n-p}}$. Much like Theorem 2.2 we now establish a lower bound on n so that w.h.p we end up (correctly) rejecting the null-hypothesis.

Theorem 3.3. *Fix a positive definite matrix $\Sigma \in \mathbb{R}^{p\times p}$. Fix parameters $\boldsymbol{\beta} \in \mathbb{R}^p$ and $\sigma^2 > 0$ and a coordinate j s.t. $\beta_j \neq 0$. Let X be a matrix whose n rows are sampled i.i.d from $\mathcal{N}(\mathbf{0}_p, \Sigma)$. Let $\boldsymbol{y}$ be a vector s.t. $y_i - (X\boldsymbol{\beta})_i$ is sampled i.i.d from $\mathcal{N}(0,\sigma^2)$. Fix $\nu \in (0,1/2)$ and $\alpha \in (0,1/2)$. Then there exist constants C_1, C_2, C_3 and C_4 such that when we run Algorithm 1 over $[X;\boldsymbol{y}]$ with parameter r w.p. $\geq 1-\alpha-\nu$ we (correctly) reject the null hypothesis using $\tilde{p}_0$ (i.e., Algorithm 1 returns* `matrix unaltered` *and we can estimate $\tilde{t}_0$ and verify that indeed $\tilde{p}_0 < \alpha \cdot e^{-\frac{r-p}{n-p}}$) provided $r \geq p + \max\left\{C_1\frac{\sigma^2(\tilde{c}_\alpha^2+\tilde{\tau}_\alpha^2)}{\beta_j^2\sigma_{\min}(\Sigma)},\ C_2\ln(1/\nu)\right\}$, and $n \geq \max\left\{r,\ C_3\frac{w^2}{\min\{\sigma_{\min}(\Sigma),\sigma^2\}},\ C_4 p\ln(1/\nu)\right\}$ where $\tilde{c}_\alpha$, $\tilde{\tau}_\alpha$ defined s.t. $\mathbf{Pr}_{X\sim T_{r-p}}[X > \tilde{c}_\alpha/e^{\frac{r-p}{n-p}}] = \mathbf{Pr}_{X\sim\mathcal{N}(0,1)}[X > \tilde{\tau}_\alpha/e^{\frac{r-p}{n-p}}] = \frac{\alpha}{2}e^{-\frac{r-p}{n-p}}$.*

3.1. Setting the Value of r, Deriving a Bound on n

Comparing the lower bound on n given by Theorem 3.3 to the bound of Theorem 2.2, we have that the data-dependent bound of $\Omega\left(\frac{(\tilde{c}_\alpha+\tilde{\tau}_\alpha)^2\sigma^2}{\beta_j^2\sigma_{\min}(\Sigma)}\right)$ should now hold for r rather than n. Yet, Theorem 3.3 also introduces an additional dependency between n and r: we require $n = \Omega(\frac{w^2}{\sigma^2} + \frac{w^2}{\sigma_{\min}(\Sigma)})$ (since otherwise we do not have $\sigma_{\min}(A) \gg w$ and Algorithm 1 might alter A before projecting it) and by definition w^2 is proportional to $\sqrt{r\ln(1/\delta)}/\epsilon$. This is precisely the focus of our discussion in this subsection. We would like to set r's value as high as possible — the larger r is, the more observations we have in RA and the better our confidence bounds (that depend on T_{r-p}) are — while satisfying $n = \Omega(\frac{\sqrt{r}}{\epsilon\cdot\min\{\sigma^2,\sigma_{\min}(\Sigma)\}})$.

Recall that if each sample point is drawn i.i.d $\boldsymbol{x} \sim \mathcal{N}(\mathbf{0}_p,\Sigma)$, then each sample $(\boldsymbol{x}_i \circ y_i)$ is sampled from $\mathcal{N}(\mathbf{0}_{p+1},\Sigma_A)$ for Σ_A defined in the proof of Theorem 3.3, that is: $\Sigma_A = \left(\begin{array}{c|c} \Sigma & \Sigma\boldsymbol{\beta} \\ \hline \boldsymbol{\beta}^\mathsf{T}\Sigma & \sigma^2+\boldsymbol{\beta}^\mathsf{T}\Sigma\boldsymbol{\beta}\end{array}\right)$. So, Theorem 3.3 gives the lower bound $r - p = \Omega\left(\frac{\sigma^2(\tilde{c}_\alpha+\tilde{\tau}_\alpha)^2}{\beta_j^2\sigma_{\min}(\Sigma)}\right)$ and the following lower bounds on n: $n \geq r$ and $n = \Omega\left(\frac{B^2(\sqrt{r\ln(1/\delta)}+\ln(1/\delta))}{\epsilon\sigma_{\min}(\Sigma_A)}\right)$, which means $r = \min\left\{n, \frac{\epsilon^2\sigma^2_{\min}(\Sigma_A)}{B^4\ln(1/\delta)}(n-\ln(1/\delta))^2\right\}$. This discussion culminates in the following corollary.

Corollary 3.4. *Denoting $\widetilde{LB_{2.2}} = \frac{\sigma^2(\tilde{c}_\alpha+\tilde{\tau}_\alpha)^2}{\beta_j^2\sigma_{\min}(\Sigma)}$, we thus conclude that if $n - p \geq \Omega\left(\widetilde{LB_{2.2}}\right)$ and $n = \Omega\left(\frac{B^2\ln(1/\delta)}{\epsilon\sigma_{\min}(\Sigma_A)}\cdot\sqrt{\widetilde{LB_{2.2}}}\right)$, then the result of Theorem 3.3 holds by setting $r = \min\left\{n, \frac{\epsilon^2\sigma^2_{\min}(\Sigma_A)}{B^4\ln(1/\delta)}(n-\ln(1/\delta))^2\right\}$.*

It is interesting to note that when we know Σ_A, we also have a bound on B. Recall Σ_A, the variance of the Gaussian $(\boldsymbol{x}\circ y)$. Since every sample is an independent draw from $\mathcal{N}(\mathbf{0}_{p+1},\Sigma_A)$ then we have an upper bound of $B^2 \leq \log(np)\sigma_{\max}(\Sigma_A)$. So our lower bound on n (using $\kappa(\Sigma_A)$ to denote the condition number of Σ_A) is given by $n \geq \max\left\{\Omega\left(\widetilde{LB_{2.2}}\right), \tilde{\Omega}\left(\frac{\kappa(\Sigma_A)\ln(1/\delta)}{\epsilon}\cdot\sqrt{\widetilde{LB_{2.2}}}\right)\right\}$. Observe, overall this result is similar in nature to many other results in differentially private learning (Bassily et al., 2014) which are of the form "without privacy, in order to achieve a total loss of $\leq \eta$ we have a sample complexity bound of some N_η; and with differential privacy the sample complexity increases to $N_\eta + \Omega(\sqrt{N_\eta}/\epsilon)$." However, there's a subtlety here worth noting. $\widetilde{LB_{2.2}}$ is proportional to $\frac{1}{\sigma_{\min}(\Sigma_A)}$ but not to $\kappa(\Sigma_A) = \frac{\sigma_{\max}(\Sigma_A)}{\sigma_{\min}(\Sigma_A)}$. The additional dependence on $\sigma_{\max}$ follows from the fact that differential privacy adds noise proportional to the upper bound on the norm of each row.

4. Projected Ridge Regression

We now turn to deal with the case that our matrix does not pass the if-condition of Algorithm 1. In this case, the matrix is appended with a $d\times d$-matrix which is $wI_{d\times d}$. Denoting $A' = \begin{bmatrix} A \\ w\cdot I_{d\times d}\end{bmatrix}$ we have that the algorithm's output is RA'. Similarly to before, we are going to denote $d = p+1$ and decompose $A = [X;\boldsymbol{y}]$ with $X\in\mathbb{R}^{n\times p}$ and $\boldsymbol{y}\in\mathbb{R}^n$, with the standard assumption of $\boldsymbol{y} = X\boldsymbol{\beta}+\boldsymbol{e}$ and e_i sampled i.i.d from $\mathcal{N}(0,\sigma^2)$. We now need to introduce some additional notation. We denote the appended matrix and vectors X' and $\boldsymbol{y}'$ s.t. $A' = [X';\boldsymbol{y}']$. And so, using the output RA' of Algorithm 1, we solve the linear regression problem derived from $\frac{1}{\sqrt{r}}RX'$ and $\frac{1}{\sqrt{r}}R\boldsymbol{y}'$. I.e., we set

$$\begin{aligned}\boldsymbol{\beta}' &= (X'^\mathsf{T}R^\mathsf{T}RX')^{-1}(RX')^\mathsf{T}(R\boldsymbol{y}') \\ \boldsymbol{\zeta}' &= \tfrac{1}{\sqrt{r}}(R\boldsymbol{y}' - RX'\boldsymbol{\beta}')\end{aligned} \tag{5}$$

Sarlos' results (2006) regarding the Johnson Lindenstrauss transform give that, when R has sufficiently many rows, solving the latter optimization problem gives a good approximation for the solution of the optimization problem $\boldsymbol{\beta}^R = \arg\min_{\boldsymbol{z}}\|\boldsymbol{y}' - X'\boldsymbol{z}\|^2 = \arg\min_{\boldsymbol{z}}\left(\|\boldsymbol{y}-X\boldsymbol{z}\|^2 + w^2\|\boldsymbol{z}\|^2\right)$. The latter problem is known as the Ridge Regression problem. Invented in the 60s (Tikhonov, 1963; Hoerl & Kennard, 1970), Ridge Regression is often motivated from the perspective of penalizing linear vectors whose coefficients are too large. It is also often applied in the case where X doesn't have full rank or is close to not having full-rank: one can show that the minimizer $\boldsymbol{\beta}^R = (X^\top X + w^2 I_{p\times p})^{-1}X^\top \boldsymbol{y}$ is the unique solution of the Ridge Regression problem and that the RHS is always well-defined.

While the solution of the Ridge Regression problem might have smaller risk than the OLS solution, it is not known how to derive t-values and/or reject the null hypothesis under Ridge Regression (except for using X to manipulate $\boldsymbol{\beta}^R$ back into $\hat{\boldsymbol{\beta}} = (X^\top X)^{-1}X^\top \boldsymbol{y}$ and relying on OLS). In fact, prior to our work there was no need for such analysis! For confidence intervals one could just use the standard OLS, because access to X and $\boldsymbol{y}$ was given.

Therefore, much for the same reason, we are unable to derive t-values under projected Ridge Regression.[7] Clearly, there are situations where such confidence bounds simply cannot be derived.Nonetheless, under additional assumptions about the data, our work can give confidence intervals for β_j, and in the case where the interval doesn't intersect the origin — assure us that $sign(\beta'_j) = sign(\beta_j)$ w.h.p. This is detailed in the supplementary material.

To give an overview of our analysis, we first discuss a model where $\boldsymbol{e} = \boldsymbol{y} - X\boldsymbol{\beta}$ is fixed (i.e., the data is fixed and the algorithm is the sole source of randomness), and prove that in this model $\boldsymbol{\beta}'$ is as an approximation to $\hat{\boldsymbol{\beta}}$.

Theorem 4.1. *Fix $X \in \mathbb{R}^{n\times p}$ and $\boldsymbol{y} \in \mathbb{R}$. Define $\hat{\boldsymbol{\beta}} = X^+\boldsymbol{y}$ and $\zeta = (I - XX^+)\boldsymbol{y}$. Let $RX' = M'$ and $R\boldsymbol{y}'$ denote the result of applying Algorithm 1 to the matrix $A = [X;\boldsymbol{y}]$ when the algorithm appends the data with a $w\cdot I$ matrix. Fix a coordinate j and any $\alpha \in (0,1/2)$. When computing $\boldsymbol{\beta}'$ and $\boldsymbol{\zeta}'$ as in (5), we have that w.p. $\geq 1 - \alpha$ it holds that $\hat{\beta}_j \in \left(\beta'_j \pm c'_\alpha \|\boldsymbol{\zeta}'\|\sqrt{\frac{r}{r-p}\cdot (M'^\top M')^{-1}_{j,j}}\right)$ where c'_α denotes the number such that $(-c'_\alpha, c'_\alpha)$ contains $1-\alpha$ mass of the T_{r-p}-distribution.*

However, our goal remains to argue that β'_j serves as a good approximation for β_j. To that end, we combine the standard OLS confidence interval — which says that w.p. $\geq 1-\alpha$ over the randomness of picking $\boldsymbol{e}$ in the homoscedastic model we have $|\beta_j - \hat{\beta}_j| \leq c_\alpha \|\boldsymbol{\zeta}\|\sqrt{\frac{(X^\top X)^{-1}_{j,j}}{n-p}}$ — with the confidence interval of Theorem 4.1 above, and denoting $I = c_\alpha \frac{\|\boldsymbol{\zeta}\|}{\sqrt{n-p}}\sqrt{(X^\top X)^{-1}_{j,j}} + c'_\alpha \frac{\|\boldsymbol{\zeta}'\|}{\sqrt{r-p}}\sqrt{r(M'^\top M')^{-1}_{j,j}}$ we have that $\mathbf{Pr}[|\beta'_j - \beta_j| = O(I)] \geq 1-\alpha$. And so, in summary, in Section C we give conditions under which the length of the interval I is dominated by the $c'_\alpha \frac{\|\boldsymbol{\zeta}'\|}{\sqrt{r-p}}\sqrt{r(M'^\top M')^{-1}_{j,j}}$ factor derived from Theorem 4.1.

5. Confidence Intervals for "Analyze Gauss"

In this section we analyze the "Analyze Gauss" algorithm of Dwork et al (2014). Algorithm 2 works by adding random Gaussian noise to $A^\top A$, where the noise is symmetric with each coordinate above the diagonal sampled i.i.d from $\mathcal{N}(0,\Delta^2)$ with $\Delta^2 = O\left(B^4 \frac{\log(1/\delta)}{\epsilon^2}\right)$. Using the same notation for a sub-matrix of A as $[X;\boldsymbol{y}]$ as before, we denote the output of Algorithm 2 as $\left(\begin{array}{c|c} \widetilde{X^\top X} & \widetilde{X^\top \boldsymbol{y}} \\ \hline \widetilde{\boldsymbol{y}^\top X} & \widetilde{\boldsymbol{y}^\top \boldsymbol{y}} \end{array}\right)$. Thus, we approximate $\boldsymbol{\beta}$ and $\|\boldsymbol{\zeta}\|$ by $\widetilde{\boldsymbol{\beta}} = \left(\widetilde{X^\top X}\right)^{-1}\widetilde{X^\top \boldsymbol{y}}$ and $\widetilde{\|\boldsymbol{\zeta}\|^2} = \widetilde{\boldsymbol{y}^\top \boldsymbol{y}} - 2\widetilde{\boldsymbol{y}^T X}\,\widetilde{\boldsymbol{\beta}} + \widetilde{\boldsymbol{\beta}}^\top \widetilde{X^\top X}\,\widetilde{\boldsymbol{\beta}}$ resp. We now argue that it is possible to use $\widetilde{\beta}_j$ and $\widetilde{\|\boldsymbol{\zeta}\|^2}$ to get a confidence interval for β_j under certain conditions.

Theorem 5.1. *Fix $\alpha, \nu \in (0, \frac{1}{2})$. Assume that there exists $\eta \in (0,\frac{1}{2})$ s.t. $\sigma_{\min}(X^\top X) > \Delta\sqrt{p\ln(1/\nu)}/\eta$. Under the homoscedastic model, given $\boldsymbol{\beta}$ and σ^2, if we assume also that $\|\boldsymbol{\beta}\| \leq B$ and $\|\hat{\boldsymbol{\beta}}\| = \|(X^\top X)^{-1}X^\top \boldsymbol{y}\| \leq B$, then w.p. $\geq 1-\alpha-\nu$ it holds that $\left|\beta_j - \widetilde{\beta}_j\right|$ is at most*

$$O\Big(\rho\cdot\sqrt{\left(\widetilde{X^\top X}^{-1}_{j,j} + \Delta\sqrt{p\ln(1/\nu)}\cdot \widetilde{X^\top X}^{-2}_{j,j}\right)\ln(1/\alpha)} \\ + \Delta\sqrt{\widetilde{X^\top X}^{-2}_{j,j}\cdot \ln(1/\nu)\cdot (B\sqrt{p}+1)}\Big)$$

where ρ is w.h.p an upper bound on σ (details appear in the Supplementary material).

Note that the assumptions that $\|\boldsymbol{\beta}\| \leq B$ and $\|\hat{\boldsymbol{\beta}}\| \leq B$ are fairly benign once we assume each row has bounded l_2-norm. The key assumption is that $X^\top X$ is well-spread. Yet in the model where each row in X is sampled i.i.d from $\mathcal{N}(\mathbf{0},\Sigma)$, this assumption merely means that n is large enough — namely, that $n = \tilde{\Omega}(\frac{\Delta\sqrt{p\ln(1/\nu)}}{\eta\cdot\sigma_{\min}(\Sigma)})$.

6. Experiment: t-Values of Output

Goal. We set to experiment with the outputs of Algorithms 1 and 2. While Theorem 3.1 guarantees that computing the t-value from the output of Algorithm 1 in the `matrix unaltered` case does give a good approximation of the t-value – we were wondering if by computing the t-value directly from the output we can (a) get a good approximation of the true (non-private) t-value and (b) get the same "higher-level conclusion" of rejecting the null-hypothesis. The answers are, as ever, mixed. The two main

[7] Note: The naïve approach of using RX' and $R\boldsymbol{y}'$ to interpolate RX and $R\boldsymbol{y}$ and then apply Theorem 3.1 using these estimations of RX and $R\boldsymbol{y}$ ignores the noise added from appending the matrix A into A', and therefore leads to inaccurate estimations of the t-values.

observations we do notice is that both algorithms improve as the number of examples increases, and that Algorithm 1 is more conservative then Algorithm 2.

Setting. We tested both algorithms in two settings. The first is over synthetic data. Much like the setting in Theorems 2.2 and 3.3, X was generated using $p = 3$ independent normal Gaussian features, and $\boldsymbol{y}$ was generated using the homoscedastic model. We chose $\boldsymbol{\beta} = (0.5, -0.25, 0)$ so the first coordinate is twice as big a the second but of opposite sign, and moreover, $\boldsymbol{y}$ is independent of the 3rd feature. The variance of the label is also set to 1, and so the variance of the homosedastic noise equals to $\sigma^2 = 1 - (0.5)^2 - (-0.25)^2$. The number of observations n ranges from $n = 1000$ to $n = 100000$.

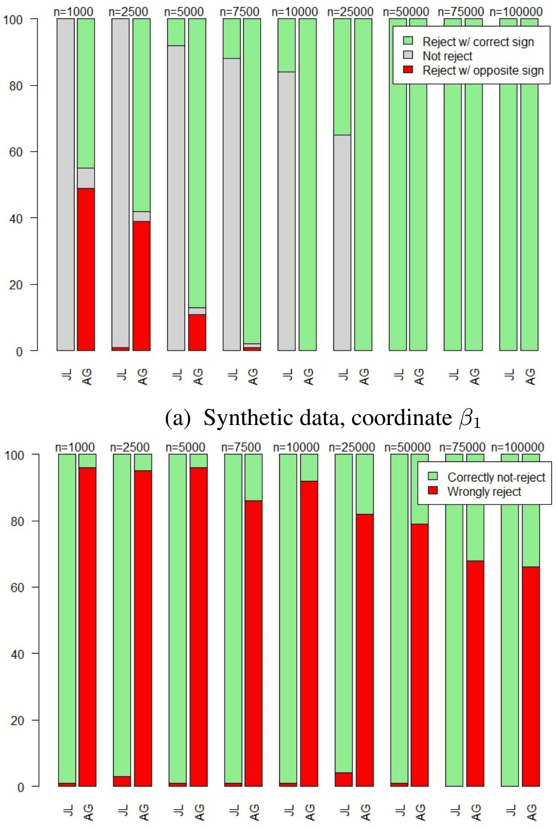
The second setting is over real-life data. We ran the two algorithms over diabetes dataset collected over ten years (1999-2008) taken from the UCI repository (Strack et al., 2014). We truncated the data to 4 attributes: sex (binary), age (in buckets of 10 years), number medications (numeric, 0-100), and a diagnosis (numeric, 0-1000). Naturally, we added a 5^{th} column of all-1 (intercept). Omitting any entry with missing or non-numeric values on these nine attributes we were left with $N = 91842$ entries, which we shuffled and fed to the algorithm in varying sizes — from $n = 30,000$ to $n = 90,000$. Running OLS over the entire N observation yields $\beta \approx (14.07, 0.54, -0.22, 482.59)$, and t-Values of $(10.48, 1.25, -2.66, 157.55)$.

The Algorithms. We ran a version of Algorithm 1 that uses a DP-estimation of $\sigma_{\min}$, and finds the largest r the we can use without altering the input, yet if this r is below 25 then it does alter the input and approximates Ridge regression. We ran Algorithm 2 verbatim. We set $\epsilon = 0.25$ and $\delta = 10^{-6}$. We repeated each algorithm 100 times.

Results. We plot the t-values we get from Algorithms 1 and 2 and decide to reject the null-hypothesis based on t-value larger than 2.8 (which corresponds to a fairly conservative p-value of 0.005). Not surprisingly, as n increases, the t-values become closer to their expected value – the t-value of Analyze Gauss is close to the non-private t-value and the t-value from Algorithm 1 is a factor of $\sqrt{\frac{r}{n}}$ smaller as detailed above (see after Corollary 3.2). As a result, when the null-hypothesis is false, Analyze Gauss tends to produce larger t-values (and thus reject the null-hypothesis) for values of n under which Algorithm 1 still does not reject, as shown in Figure 1a. This is exacerbated in real data setting, where its actual least singular value (≈ 500) is fairly small in comparison to its size ($N = 91842$).

However, what is fairly surprising is the case where the null-hypothesis should not be rejected — since $\beta_j = 0$ (in the synthetic case) or its non-private t-value is close to 0 (in the real-data case). Here, the Analyze Gauss' t-value approximation has fairly large variance, and we still get fairly high (in magnitude) t-values. As the result, we falsely reject the null-hypothesis based on the t-value of Analyze Gauss quite often, even for large values of n. This is shown in Figure 1b. Additional figures (including plotting the distribution of the t-value approximations) appear in the supplementary material.

The results show that t-value approximations that do not take into account the inherent randomness in the DP-algorithms lead to erroneous conclusions. One approach would be to follow the more conservative approach we advocate in this paper, where Algorithm 1 may allow you to get true approximation of the t-values and otherwise reject the null-hypothesis only based on the confidence interval (of Algorithm 1 or 2) not intersecting the origin. Another approach, which we leave as future work, is to replace the T-distribution with a new distribution, one that takes into account the randomness in the estimator as well. This, however, has been an open and long-standing challenge since the first works on DP and statistics (see (Vu & Slavkovic, 2009; Dwork & Lei, 2009)) and requires we move into non-asymptotic hypothesis testing.



(a) Synthetic data, coordinate β_1

(b) Synthetic data, coordinate β_3

Figure 1. Correctly and Wrongly Rejecting the Null-Hypothesis

Acknowledgements

The bulk of this work was done when the author was a postdoctoral fellow at Harvard University, supported by NSF grant CNS-123723; and also an unpaid collaborator on NSF grant 1565387. The author wishes to wholeheartedly thank Prof. Salil Vadhan, for his tremendous help in shaping this paper. The author would also like to thank Prof. Jelani Nelson and the members of the "Privacy Tools for Sharing Research Data" project at Harvard University (especially James Honaker, Vito D'Orazio, Vishesh Karwa, Prof. Kobbi Nissim and Prof. Gary King) for many helpful discussions and suggestions; as well as Abhradeep Thakurta for clarifying the similarity between our result. Lastly the author thanks the anonymous referees for many helpful suggestions in general and for a reference to (Ullman, 2015) in particular.

References

Agresti, A. and Finlay, B. *Statistical Methods for the Social Sciences*. Pearson P. Hall, 2009.

Bassily, R., Smith, A., and Thakurta, A. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *FOCS*, 2014.

Blocki, J., Blum, A., Datta, A., and Sheffet, O. The Johnson-Lindenstrauss transform itself preserves differential privacy. In *FOCS*, 2012.

Chaudhuri, Kamalika and Hsu, Daniel J. Convergence rates for differentially private statistical estimation. In *ICML*, 2012.

Chaudhuri, Kamalika, Monteleoni, Claire, and Sarwate, Anand D. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12, 2011.

Duchi, John C., Jordan, Michael I., and Wainwright, Martin J. Local privacy and statistical minimax rates. In *FOCS*, pp. 429–438, 2013.

Dwork, C. and Lei, J. Differential privacy and robust statistics. In *STOC*, 2009.

Dwork, Cynthia, Kenthapadi, Krishnaram, McSherry, Frank, Mironov, Ilya, and Naor, Moni. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, 2006a.

Dwork, Cynthia, Mcsherry, Frank, Nissim, Kobbi, and Smith, Adam. Calibrating noise to sensitivity in private data analysis. In *TCC*, 2006b.

Dwork, Cynthia, Talwar, Kunal, Thakurta, Abhradeep, and Zhang, Li. Analyze gauss - optimal bounds for privacy preserving principal component analysis. In *STOC*, 2014.

Dwork, Cynthia, Su, Weijie, and Zhang, Li. Private false discovery rate control. *CoRR*, abs/1511.03803, 2015.

Hoerl, A. E. and Kennard, R. W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

Kasiviswanathan, S., Lee, H., Nissim, K., Raskhodnikova, S., and Smith, A. What can we learn privately? In *FOCS*, 2008.

Kifer, Daniel, Smith, Adam D., and Thakurta, Abhradeep. Private convex optimization for empirical risk minimization with applications to high-dimensional regression. In *COLT*, 2012.

Laurent, B. and Massart, P. Adaptive estimation of a quadratic functional by model selection. *The Annals of Statistics*, 28(5), 10 2000.

Ma, E. M. and Zarowski, Christopher J. On lower bounds for the smallest eigenvalue of a hermitian positive-definite matrix. *IEEE Transactions on Information Theory*, 41(2), 1995.

Muller, Keith E. and Stewart, Paul W. *Linear Model Theory: Univariate, Multivariate, and Mixed Models*. John Wiley & Sons, Inc., 2006.

Pilanci, M. and Wainwright, M. Randomized sketches of convex programs with sharp guarantees. In *ISIT*, 2014a.

Pilanci, Mert and Wainwright, Martin J. Iterative hessian sketch: Fast and accurate solution approximation for constrained least-squares. *CoRR*, abs/1411.0347, 2014b.

Rao, C. Radhakrishna. *Linear statistical inference and its applications*. Wiley, 1973.

Rogers, Ryan M., Vadhan, Salil P., Lim, Hyun-Woo, and Gaboardi, Marco. Differentially private chi-squared hypothesis testing: Goodness of fit and independence testing. In *ICML*, pp. 2111–2120, 2016.

Rudelson, Mark and Vershynin, Roman. Smallest singular value of a random rectangular matrix. *Comm. Pure Appl. Math*, pp. 1707–1739, 2009.

Sarlós, T. Improved approx. algs for large matrices via random projections. In *FOCS*, 2006.

Sheffet, O. Private approximations of the 2nd-moment matrix using existing techniques in linear regression. *CoRR*, abs/1507.00056, 2015. URL http://arxiv.org/abs/1507.00056.

Smith, Adam D. Privacy-preserving statistical estimation with optimal convergence rates. In *STOC*, pp. 813–822, 2011.

Strack, B., DeShazo, J., Gennings, C., Olmo, J., Ventura, S., Cios, K., and Clore, J. Impact of HbA1c measurement on hospital readmission rates: Analysis of 70,000 clinical database patient records. *BioMed Research International*, 2014:11 pages, 2014.

Tao, T. *Topics in Random Matrix Theory*. American Mathematical Soc., 2012.

Thakurta, Abhradeep and Smith, Adam. Differentially private feature selection via stability arguments, and the robustness of the lasso. In *COLT*, 2013.

Tikhonov, A. N. Solution of incorrectly formulated problems and the regularization method. *Soviet Math. Dokl.*, 4, 1963.

Uhler, Caroline, Slavkovic, Aleksandra B., and Fienberg, Stephen E. Privacy-preserving data sharing for genome-wide association studies. *Journal of Privacy and Confidentiality*, 2013. Available at: http://repository.cmu.edu/jpc/vol5/iss1/6.

Ullman, J. Private multiplicative weights beyond linear queries. In *PODS*, 2015.

Vu, D. and Slavkovic, A. Differential privacy for clinical trial data: Preliminary evaluations. In *ICDM*, 2009.

Wang, Yue, Lee, Jaewoo, and Kifer, Daniel. Differentially private hypothesis testing, revisited. *CoRR*, abs/1511.03376, 2015.

Xi, B., Kantarcioglu, M., and Inan, A. Mixture of gaussian models and bayes error under differential privacy. In *CODASPY*. ACM, 2011.

Zhou, S., Lafferty, J., and Wasserman, L. Compressed regression. In *NIPS*, 2007.