Differentially private vector aggreation in the case of multivariate gaussian data

Tim Sehested Poulsen

January 15, 2023

> datasets -¿ dataset

> Write introduction, mention relation to average vector

1 Preliminaries

1.1 Definitions

The dataset is denoted by X, where $X \in \mathbb{R}^{n \times d}$. I have that n denotes the number of entries in the dataset and d is the number of dimensions of the dataset. I will throughout the report refer to a single entry of the dataset as x_i and a single dimension of the dataset as $X^{(j)}$, and therefore $x_i^{(j)}$ denotes the j'th dimension of the i'th entry.

Differential privacy is the heuristic of releasing a database statistic whilst limiting the impact of any one entry. It builds on the intuition that computing a statistics on a private dataset should not reveal any sensitive information about any one individual as long as that individual has little to no effect on the outcome. Differential privacy has multiple slightly different formal definitions, one such is (ε, δ)-Differential Privacy refered to as (ε, δ)-DP which will be introduced later on. A prerequisite for almost all of the different differential privacy definitions relies on the concept of neighbouring dataset.

Definition 1.1 (Neighbouring dataset [6]). *Two dataset $X, X' \in \mathbb{R}^{n \times d}$ are said to be neigbouring if they differ in at most a single entry. Neighbouring dataset are denoted with the relation $X \sim X'$ and defined as followed*

$$X \sim X' \iff |\{i \in \mathbb{N} \mid i \leq n \wedge x_i \neq x'_i\}| \leq 1$$

Definition 1.2 (Sensitivity [7]). *Let $f(X) : \mathbb{R}^{n \times d} \to \mathbb{R}^k$ be a function. The l_p-sensitivity of f is the maximal possible l_p-norm of the difference between the output of f on two neighbouring dataset. We denote the sensitivity as*

$$\Delta_p(f) = \max_{X \sim X'} \|f(X) - f(X')\|_p$$

and then the total l_2-sensitivity is then

Throughout the report I will only be working with l_2-sensitivity and will just denote this as $\Delta(f)$ for ease of notation.

Definition 1.3 ((ε, δ)-Differential Privacy [6]). *A randomized algorithm $\mathcal{M} : \mathbb{R}^{n \times d} \to \mathcal{R}$ is (ε, δ)-differentially private if for all possible subsets of outputs $S \subseteq \mathcal{R}$ and all pairs of neighbouring dataset $X \sim X'$ we have that*

$$\Pr\left[M(X) \in S\right] \le e^\varepsilon \cdot \Pr\left[M(X') \in S\right] + \delta$$

Theorem 1 ((ε, δ)-DP under post-processing [7]). *Let $\mathcal{M} : \mathbb{R}^{n \times d} \to \mathcal{R}$ be an (ε, δ)-DP algorithm. Let $f : \mathcal{R} \to \mathcal{R}'$ be an arbitrary mapping, then $f \circ \mathcal{M} : \mathbb{R}^{n \times d} \to \mathcal{R}'$ is (ε, δ)-DP .*

Proof

Fix any pair of neighbouring datasets $X \sim X'$ and let $S \subseteq \mathcal{R}'$ be an arbitrary event. We then define $T = \{r \in \mathcal{R} \mid f(r) \in S\}$. We thus have that

$$\begin{aligned}
\Pr\left[f(\mathcal{M}(X)) \in S\right] &= \Pr\left[\mathcal{M}(X) \in T\right] \\
&\le e^\varepsilon \cdot \Pr\left[\mathcal{M}(X') \in T\right] + \delta = e^\varepsilon \cdot \Pr\left[f(\mathcal{M}(X')) \in S\right] + \delta
\end{aligned}$$

$\blacksquare$

Error Measure As this report concerns itself exclusively with the sum of entries in a dataset, error will be defined as the expected squared l_2-norm between the true sum and the output of a randomized algorithm. So let $X \in \mathbb{R}^{n \times d}$ be the dataset and $f(X) = \sum_i^n x_i$ be the true sum of all entries. The error of a randomized algorithm $M : \mathbb{R}^{n \times d} \to \mathbb{R}^d$ which estimates $f(X)$ is then

$$\text{Err}(M) := \mathbb{E}\left[\|M(X) - f(X)\|^2\right]$$

extra

1.2 Quadratic forms of random variables

write cohesive text here, and decide what to keep

Lemma 1.1. *Let $X \sim \mathcal{N}(\mu, \sigma^2)$ be a gaussian random variable and let $r \in \mathbb{R}$ be a constant. We then have that*

$$\begin{aligned}
rX &\sim \mathcal{N}(\mu, (r\sigma)^2) \\
X - r &\sim \mathcal{N}(\mu - r, \sigma^2)
\end{aligned}$$

Quadratics of random variables have been well studied [2, 11], specially in the case of multivariate gaussian varaibles [8, 11]. Even more research has been done in evaluating the CDF of these quadratic forms for Gaussian random vectors [4, 9].

Lemma 1.2 (Expectation of a quadratic random variable [2])**.** *Let X be a d-dimensional random vector with expected value $\mathbb{E}[X] = \boldsymbol{\mu_X}$ and covariance matrix $Var[X] = \boldsymbol{\Sigma_X}$. Let also A be a constant $d \times d$ symmetric matrix, then*

$$\mathbb{E}\left[X^T A X\right] = tr\left(A\boldsymbol{\Sigma_X}\right) + \boldsymbol{\mu}^T A \boldsymbol{\mu}$$

Proof
Blah blah

$$\begin{aligned}
\mathbb{E}\left[X^T A X\right] &= \mathrm{tr}\left(\mathbb{E}\left[X^T A X\right]\right) = \mathbb{E}\left[\mathrm{tr}\left(X^T A X\right)\right] \\
&= \mathbb{E}\left[\mathrm{tr}\left(A X X^T\right)\right] = \mathrm{tr}\left(A\mathbb{E}\left[X X^T\right]\right) = \mathrm{tr}\left(A\left(\mathrm{Var}\left[X\right] + \boldsymbol{\mu}\boldsymbol{\mu}^T\right)\right) \\
&= \mathrm{tr}\left(A\boldsymbol{\Sigma}\right) + \mathrm{tr}\left(A\boldsymbol{\mu}\boldsymbol{\mu^T}\right) = \mathrm{tr}\left(A\boldsymbol{\Sigma}\right) + \boldsymbol{\mu}^T A \boldsymbol{\mu}
\end{aligned}$$

blah ∎

Corollary 1.1. *Let $X \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma_X})$ be a d-dimensional gaussian vector with expected value $\mathbf{0}$, and let σ_j^2 denote the variance of the j'th dimension where $1 \leq j \leq d$. By theorem 1.2 we have that the expected l_2-norm of such a vector is given by*

$$\mathbb{E}\left[\|X\|^2\right] = tr(\boldsymbol{\Sigma_X}) = \sum_{j=1}^{d} \sigma_j^2$$

2 Algorithms

2.1 The Gaussian Mechanism

One of the most fundamental algorithms for achieving (ε, δ)-DP is the Gaussian Mechanism [7]. It computes the real value of a statistic, where the l_2-sensitivity is known. That is it produces a (ε, δ)-DP estimate of a function $g : \mathbb{R}^{n \times d} \to \mathbb{R}^d$ where the l_2-sensitivity $\Delta(g)$ is known. It does so by computing the value of $g(X)$ and then adding noise to each dimension drawn from the normal distribution $\mathcal{N}(0, \sigma_{\varepsilon,\delta}^2)$. This can be seen as adding a noise vector η which is then distributed according to the multivariate normal distribution $\mathcal{N}(\mathbf{0}, \sigma_{\varepsilon,\delta}^2 I)$. Pseudo code for the algorithm can be seen in Algorithm 1.

Algorithm 1 The Gaussian Mechanism

   Input
      $\sigma_{\varepsilon,\delta}$ Standard deviation required to achieve (ε, δ)-DP
      $X \in \mathbb{R}^{n \times d}$ Dataset
   Output
      (ε, δ)-DP estimate of $g(X)$
   for $j \in [d]$ **do**
      $\eta_j \leftarrow$ sample from $\mathcal{N}(0, \sigma_{\varepsilon,\delta}^2)$
   end for
   return $g(X) + \boldsymbol{\eta}$

The algorithm quite intuitively prodcues error which is purely given by the norm of the noise added, and the expected error can be calculated to be

$$\mathbb{E}\left[\|(g(X) + \boldsymbol{\eta}) - g(X)\|^2\right] = \mathbb{E}\left[\|\boldsymbol{\eta}\|^2\right]$$

Which by corollary 1.1 is

$$\mathbb{E}\left[\|\eta\|^2\right] = \sum_i^d \sigma_{\varepsilon,\delta}^2 = d \cdot \sigma_{\varepsilon,\delta}^2 \tag{1}$$

It is apparent that the main difficulty of the mechanism lies in determining a $\sigma_{\varepsilon,\delta}$ which achieves (ε, δ)-DP , and preferably the smallest such one.

The following theorem was initially proven

Theorem 2. [7] *Let $g : \mathbb{R}^{n \times d} \to \mathbb{R}^d$ be an arbitrary d-dimensional function with l_2-sensitvity $\Delta(g) = \max_{X \sim X'} \|g(X) - g(X')\|$, and let $\varepsilon \in (0,1)$. The Gaussian Mechanism with $\sigma_{\varepsilon,\delta} = \Delta(g)\sqrt{2\ln(1.25/\delta)}/\varepsilon$ is (ε, δ)-DP .*

The proof is rather long and is therefore ommitted here. A few years later it was shown in [1] how to compute the minimal $\sigma_{\varepsilon,\delta}$.

Theorem 3. [1] *The Gaussian Mechanism is differentially private if and only if $\sigma_{\varepsilon,\delta} \geq \Delta(g) \cdot \sigma_{opt}$ where $\sigma_{opt} \in \mathbb{R}$ is the smallest value greater than 0, which satisfies*

$$\Phi\left(\frac{1}{2\sigma_{opt}} - \varepsilon\sigma_{opt}\right) - e^\varepsilon \Phi\left(-\frac{1}{2\sigma_{opt}} - \varepsilon\sigma_{opt}\right) \leq \delta \tag{2}$$

In [1] it is also shown how to compute this value, but since this is of no importance to this project it will therefore not be covered here. For the rest of the report I will only be referring to σ_{opt} as the minimal value given by equation (2).

The main downside of the Gaussian Mechanism is that it adds equal noise to all dimensions, regardless of the sensitivity in that dimension. As this report focuses on giving differentially private estimates of sums of vectors, it is natural to ask whether adding equal noise in all dimensions is optimal in this setting. Let us define the function of interest $f(X) = \sum_{i \in [n]} x_i$, for a dataset $X \in \mathbb{R}^{n \times d}$. The sensitivity of this function must be twice the largest possible norm of any vector. Therefore if the largest difference between any two neighbouring datasets in the j'th dimension can be limited as follows

$$\Delta_j := \max_{X \sim X'} |X^{(j)} - X'^{(j)}| \tag{3}$$

and we say that X and X' differ in the i'th entry we conclude that

$$\Delta(f) = \max_{X \sim X'} \|f(X) - f(X')\| = \max_{X \sim X'} \|x_i - x_i'\| = \sqrt{\sum_{j \in [d]} \Delta_j^2} = \|\boldsymbol{\Delta}\| \tag{4}$$

where $\boldsymbol{\Delta} = (\Delta_1, \Delta_2, \ldots, \Delta_d)$.
This means that by equation 1 the expected error of the Gaussian Mechanism when estimating $f(X)$ is given by

$$\mathbb{E}\left[\|\eta\|^2\right] = d \cdot \sigma_{\varepsilon,\delta}^2 = d \cdot (\Delta(f) \cdot \sigma_{opt})^2 = d \cdot \sigma_{opt}^2 \cdot \|\boldsymbol{\Delta}\|^2 \tag{5}$$

4

A logical next step would be to add noise to each dimensions such that it is proportional to the sensitivity of that dimension. This has been studied in [12] and lays the foundation for this report. Their mechanism, appropriately called the Elliptical Gaussian Mechanism works very similairly to the Gaussian Mechanism as described by algorithm 1, though instead of sampling all η_j from $\mathcal{N}(0, \sigma_{\varepsilon,\delta}^2)$, here they are instead drawn from $\mathcal{N}\left(0, \left(\sigma_{opt} \cdot \frac{\Delta_j}{b_j}\right)^2\right)$. The algorithm is described in detail in Algorithm 2

Algorithm 2 The Elliptical Gaussian Mechanism

Input

σ_{opt}	Standard deviation as defined by theorem 3
$X \in \mathbb{R}^{n \times d}$	Dataset
$\boldsymbol{b} \in \mathbb{R}^d$	Scaling vector, where $\|\boldsymbol{b}\| = 1$
$\boldsymbol{\Delta} \in \mathbb{R}^d$	Sensitivies of all dimensions

Output

(ε, δ)-DP estimate of $f(X) = \sum_{i \in [n]} x_i$

for $j \in [d]$ **do**

$\qquad \sigma_j \leftarrow \sigma_{opt} \cdot \frac{\Delta_j}{b_j}$

$\qquad \eta_j \leftarrow$ sample from $\mathcal{N}(0, \sigma_j^2)$

end for

return $f(X) + \eta$

Though the algorithm seems short and simple, as the the only thing which distinguishes it from The Gaussian Mechanism is to draw samples from $\mathcal{N}(0, \left(\sigma_{opt} \cdot \frac{\Delta_j}{b_j}\right)^2)$, it can be seen as a transformation of space. Consider transforming the data by the factor $\frac{b_j}{\Delta_j}$ in each dimension. Then you would have that each dimension is bounded

$$x_{i,j} \leq \frac{\Delta_j}{2} \implies \widehat{x}_{i,j} := x_{i,j} \cdot \frac{b_j}{\Delta_j} \leq \frac{b_j}{2}$$

and in this space the sensitivity of $f(X) = \sum_{i \in [n]} x_i$ is then

$$\Delta(f) = \sqrt{\sum_{j \in [d]} \left(\frac{b_j}{2} - \left(-\frac{b_j}{2}\right)\right)^2} = \sqrt{\sum_{j \in [d]} b_j^2} = \|\boldsymbol{b}\| = 1$$

This means that all points in X are mapped to lie withing the unit ball, and in this space one can simply add noise to each dimension drawn from $\eta_j \sim \mathcal{N}(0, \sigma_{opt})$. Then the inverse transformation $\frac{\Delta_j}{b_j}$ can then be applied to each dimension of $f(\overline{X})$ as well as η, giving a differentially private estimate of $f(X)$, because differential privacy is preserved under post processing as per Theorem 1.

The main thing that really differentiates Algorithm 2 from the normal Gaussian Mechanism is the introduction of the scaling vector $\boldsymbol{b}$. This is the scaling of how much weight should be attributed to each dimension when adding noise. It is shown how to determine the optimal values for b_j in [12], and what the expected error of the mechanism then is.

Theorem 4 (Optimality and error of the Elliptical Gaussian Mechanism [12]). *The value for b_j which minimizes the expected l_2 error $\mathbb{E}\left[\|\boldsymbol{\eta}\|^2\right]$ of Algorithm 2 is as follows*

$$b_j = \sqrt{\frac{\Delta_j}{\sum_{j \in [d]} \Delta_j}}$$

which leads the error to be

$$\mathbb{E}\left[\|\boldsymbol{\eta}\|^2\right] = \sigma_{opt}^2 \cdot \|\boldsymbol{\Delta}\|_1^2 \tag{6}$$

where $\|\cdot\|_1$ is the l_1 norm.

The proof will be ommitted here, but it is very similair to the proofs for lemma 3.1 and Theorem 5 which are in section 3.

Comparing the expected error between Algorithm 1 and Algorithm 2 we have the following ratio

$$\frac{d \cdot \sigma_{opt}^2 \cdot \|\boldsymbol{\Delta}\|^2}{\sigma_{opt}^2 \cdot \|\boldsymbol{\Delta}\|_1^2} = \left(\frac{\sqrt{d}\|\boldsymbol{\Delta}\|}{\|\boldsymbol{\Delta}\|_1}\right)^2 = \frac{d \sum_{j \in [d]} \Delta_j^2}{\left(\sum_{j \in [d]} \Delta_j\right)^2}$$

in which it can be seen that they are equal when all entries of $\boldsymbol{\Delta}$ are the same. Otherwise the error for the Elliptical Gaussian Mechanism is lower when $\boldsymbol{\Delta}$ is skewed. As argued in [12] Algorithm 2 improves Algorithm 1 by a factor in $[1, d)$.

3 Problem setup

As previously mentioned the problem investigated here consists of realeasing the sum of vectors in a dataset under differential privacy. More formally we whish to release the value of $f(X) = \sum_{i=1}^{n} x_i$ under (ε, δ)-DP .

The common factor for achieving (ε, δ)-DP in both the Gaussian Mechanism and the Elliptical Gaussian Mechanism is the knowledge that data lie within some bounds. Specifically for the Elliptical Gaussian Mechanism data is required to lie within some hyperrectangle. It is formally described by equation (3) essentially saying that there is an upper and lower bound on each dimension. This requirement is needed to know the l_2-sensitivity $\Delta(f)$ as shown in equation (4). In this project I will change this assumption and instead look at the case where each dimension is normally distributed. This means that for each $j \in [d]$ we have that $x_{i,j} \sim \mathcal{N}(\mu_j, \sigma_j^2)$. An equivalent formulation is that the data is multivariately gaussianly distributed but with no correlation between dimensions. This means that $x_i \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, where Σ is a diagonal matrix with the variance of each dimension along its diagonal. It is quite apparent that determining a limit Δ_j is impossible in this setting as the gaussian distribution is continously defined on the range $(-\infty, \infty)$. Several recent papers has combatted this by doing something called *clipping* [3, 10]. Clipping is the process of limiting the norm of any one entry to be at most a chosen threshold C. This means that every vector is transformed as such

$$\overline{x}_i := \min\left\{\frac{C}{\|x_i\|}, 1\right\} \cdot x_i$$

Clipping entries by a factor C thus means that $\Delta(f) = 2C$ as any one entry cannot have more impact on the summation than C. If the summation $f(X)$ is performed on a clipped dataset $\overline{X}$ it is equivalent to defining the summation function $\overline{f} : \mathbb{R}^{n \times d} \to \mathbb{R}^d$ as

$$\overline{f}(X) = \sum_i^n \min\left\{\frac{C}{\|x_i\|}, 1\right\} \cdot x_i$$

Then by Theorem 3 the gaussian mechanism with the function $\overline{f}$ is (ε, δ)-DP with $\sigma_{\varepsilon,\delta} = \Delta\left(\overline{f}\right)\sigma_{opt} = 2C\sigma_{opt}$. Though the mechanism is still (ε, δ)-DP it will now have a larger error when regarding the true sum $f(X) = \sum_i^n x_i$ as the actual answer. If the probability of clipping is set so low that we actually don't expect to clip any entries we can use $\overline{f}$ as an approximation of f. Say there are n points in a dataset, I will thus set the probability of clipping to be less than $\frac{1}{n}$ and get that

$$\mathbb{E}\left[\|\left(\overline{f}(X) + \eta\right) - f(X)\|^2\right] \approx \mathbb{E}\left[\|\left(f(X) + \eta\right) - f(X)\|^2\right] = \mathbb{E}\left[\|\eta\|^2\right]$$

In this setting, we again have the intuition that adding equal noise in all dimensions is non optimal, and instead the noise in a dimension should be proportional to the variance σ_j^2 in that dimension. In a desire to achieve similair results to that of the Elliptical Gaussian Mechanism, just with normally distributed data, I will use somewhat the same approach to achieve (ε, δ)-DP . They achieve (ε, δ)-DP by finding a transformation of points such that they all lie within the unit ball centered at the origin. There does not exist such a transformation which is linear (I think I mean Continous/Homeomorphic), as there will always be a non-zero probability of observing points outside the unit ball. Instead I will introduce the constraint that the expected norm of vectors after the transformation should be 1. I whish to find a scaling of each dimension b_j s.t.

$$\widehat{x}_i := x_i \odot \boldsymbol{b} \tag{7}$$
$$\mathbb{E}\left[\|\widehat{x}\|\right] = 1 \iff \mathbb{E}\left[\|\widehat{x}\|^2\right] = 1 \tag{8}$$

where $\boldsymbol{b} = (b_1, b_2, \ldots, b_d)$, and $\odot$ is the element-wise product. Then $\overline{f}$ can be computed on this transformed dataset $\widehat{X}$, where the probability of clipping is less than $\frac{1}{n}$. As Theorem 1 shows, (ε, δ)-DP is preserved under post processing, we can therefore add noise to each coordinate drawn from $\mathcal{N}(0, (2C\sigma_{opt})^2)$ in the transformed space to achieve (ε, δ)-DP . The transformation back to the original space is then done by multiplying each dimension with b_j^{-1}, and due to the linearity of transformation this is also done to the noise added. We end up with the following noise vector being added

$$\boldsymbol{\eta} = \left(b_1^{-1}\eta_1, b_2^{-1}\eta_2, \ldots, b_d^{-1}\eta_d\right)$$

in which all $\eta_j \sim \mathcal{N}(0, (2C\sigma_{opt})^2)$, and then by lemma 1.1 we have $b_j^{-1}\eta_j \sim \mathcal{N}(0, (b_j^{-1} \cdot 2C\sigma_{opt})^2)$. As the error is approximately given by $\|\boldsymbol{\eta}\|^2$ we have due to Corollary 1.1 that the expected error is

$$\mathbb{E}\left[\|\boldsymbol{\eta}\|^2\right] = \sum_{j=1}^d \left(b_j^{-1} \cdot 2C\sigma_{opt}\right)^2 = (2C\sigma_{opt})^2 \sum_{j=1}^d b_j^{-2} \tag{9}$$

Algorithm 3 The Elliptical Gaussian Mechanism for Gaussian data

---

Input

$\quad$ $\sigma_{opt} \in \mathbb{R}$ $\qquad$ Standard deviation as defined by Theorem 3

$\quad$ $\boldsymbol{\sigma} \in \mathbb{R}^d$ $\qquad$ Standard deviations of all dimension

$\quad$ $X \in \mathbb{R}^{n \times d}$ $\qquad$ Dataset, where $x_{i,j} \sim \mathcal{N}(\mu_j, \sigma_j^2)$

$\quad$ $\boldsymbol{b} \in \mathbb{R}^d$ $\qquad$ Scaling vector, s.t. $\mathbb{E}\left[\|x_i \odot \boldsymbol{b}\|^2\right] = 1$

$\quad$ $p_C \in \mathbb{R}$ $\qquad$ Probability of clipping points

Output

$\quad$ (ε, δ)-DP estimate of $f(X)$

$\widehat{\boldsymbol{\sigma}} \leftarrow \boldsymbol{\sigma} \odot \boldsymbol{b}$

$C \leftarrow$ Minimal C satisfying $\Pr\left[\|\widehat{x}\| > C\right] \leq p_C$, for $\widehat{x} \sim \mathcal{N}(\boldsymbol{0}, \widehat{\boldsymbol{\sigma}})$

$T \leftarrow \boldsymbol{0}$

for $i \in [n]$ **do**

$\quad$ $\widehat{x}_i \leftarrow x_i \odot \boldsymbol{b}$

$\quad$ $T \leftarrow T + \min\left(\frac{C}{\|\widehat{x}_i\|}, 1\right) \cdot \widehat{x}_i$

end for

for $j \in [d]$ **do**

$\quad$ $T_j \leftarrow T_j \cdot b_j^{-1}$

$\quad$ $\sigma_{\varepsilon,\delta,j} \leftarrow 2C\sigma_{opt} \cdot b_j^{-1}$

$\quad$ $\eta_j \leftarrow$ sample from $\mathcal{N}(0, \sigma_{\varepsilon,\delta,j}^2)$

end for

return $T + \boldsymbol{\eta}$

The pseudo code for the mechanism is given in Algorithm 3. At first glance it seems quite different from Algorithm 2 despite it being very similiar. The most notable thing here is that the transformation has to be performed on the dataset, as it is important which points are being clipped, in order to ensure that the mechanism is (ε, δ)-DP . In Algorithm 2 only the inverse transformation, b_j^{-1}, is being applied on the noise to be added.

Now we are back to a very similiar problem to the one solved in [12]. Minimize the error described in equation (9) under the constraint given in equation (8). The constraint can be given as

$$\mathbb{E}\left[\|x_i \odot \boldsymbol{b}\|^2\right] = \sum_{j \in [d]} (b_j \sigma_j)^2 = 1 \tag{10}$$

due to Corollary 1.1.

This leads to the following Lemma

Lemma 3.1. *Let $X \in \mathbb{R}^{n \times d}$ be a dataset where each dimension is independently gaussianly distributed, i.e. $x_{i,j} \sim \mathcal{N}(\mu_j, \sigma_j^2)$. Then the expected error of Algorithm 3 is minimized when*

$$b_j = \frac{1}{\sqrt{\sigma_j}\sqrt{\sum_{i=1}^d \sigma_i}}$$

$\quad$ *Proof*

Using lagrangian multipliers we find the local maxima or minia of the function subject to equality constraints. To do so we construct the lagrangian function $\mathcal{L} : \mathbb{R}^{d+1} \to \mathbb{R}$ from the optimization problem given by equation (9) and the constraint given by (10) for ease of notation we define $\sigma_{\varepsilon,\delta} := 2C\sigma_{opt}$.

$$\mathcal{L}(\boldsymbol{b}, \lambda) = \sigma_{\varepsilon,\delta}^2 \sum_{j \in [d]} b_j^{-2} + \lambda \left(\sum_{j=1}^{d} (\sigma_j b_j)^2 - 1 \right)$$

We then find the stationary points of it, by setting the derivative of it to $\boldsymbol{0}$. The derivative with respect to b_j is

$$\frac{\partial \mathcal{L}}{\partial b_j}(\boldsymbol{b}, \lambda) = \frac{\partial}{\partial b_j} \left(\sigma_{\varepsilon,\delta}^2 \cdot b_j^{-2} + \lambda (\sigma_j b_j)^2 \right) = -2\sigma_{\varepsilon,\delta}^2 b_j^{-3} + 2\lambda \sigma_j^2 b_j$$

I then solve $\frac{\partial \mathcal{L}}{\partial b_j}(\boldsymbol{b}, \lambda) = 0$ for b_j

$$- 2\sigma_{\varepsilon,\delta}^2 b_j^{-3} + 2\lambda \sigma_j^2 b_j = 0 \iff \lambda \sigma_j^2 b_j = \sigma_{\varepsilon,\delta}^2 b_i^{-3} \tag{11}$$

$$\iff b_j^4 = \frac{\sigma_{\varepsilon,\delta}^2}{\lambda \sigma_j^2} \iff b_j = \frac{\sqrt{\sigma_{\varepsilon,\delta}}}{\lambda^{\frac{1}{4}} \sqrt{\sigma_j}} \tag{12}$$

I now have the last partial derivative $\frac{\partial \mathcal{L}}{\partial \lambda}(\boldsymbol{b}, \lambda) = 0$ which I solve for λ using the previous expression for b_j.

$$\frac{\partial \mathcal{L}}{\partial \lambda}(\boldsymbol{b}, \lambda) = \sum_{j=1}^{d} (\sigma_j b_j)^2 - 1$$

$$\sum_{j=1}^{d} (\sigma_j b_j)^2 - 1 = 0 \iff \sum_{j=1}^{d} \sigma_j^2 \left(\frac{\sigma_{\varepsilon,\delta}}{\sqrt{\lambda} \sigma_j} \right) = 1 \iff$$

$$\frac{\sigma_{\varepsilon,\delta}}{\sqrt{\lambda}} \sum_{j=1}^{d} \frac{\sigma_j^2}{\sigma_j} = 1 \iff \sigma_{\varepsilon,\delta} \sum_{j=1}^{d} \sigma_j = \sqrt{\lambda}$$

Inserting back into equation (12)

$$b_j = \frac{\sqrt{\sigma_{\varepsilon,\delta}}}{\lambda^{\frac{1}{4}} \sqrt{\sigma_j}} = \frac{\sqrt{\sigma_{\varepsilon,\delta}}}{\sqrt{\sigma_{\varepsilon,\delta} \sum_{i=1}^{d} \sigma_i} \sqrt{\sigma_j}} = \frac{1}{\sqrt{\sigma_j} \sqrt{\sum_{i=1}^{d} \sigma_i}}$$

> show that this stationary point is a minimum

Giving us the value of b_j which minimizes the expected error. ∎

Now to evaluate the expected error of the mechanism, which is given by equation (9), it is highly important important to determine the value C. As C should be given by the minimal value which satisfies the inequality

$$\Pr \left[\|x_i \odot \boldsymbol{b}\| > C \right] = \Pr \left[\|x_i \odot \boldsymbol{b}\|^2 > C^2 \right] < \frac{1}{n}$$

again where n denotes the number of points in the dataset. As x_i is a multivariate gaussian distribution, so is $x_i \odot \boldsymbol{b}$ by lemma 1.1, and $\|x_i \odot \boldsymbol{b}\|^2$ is the sum of independent gaussian variables squared. Such a sum is distributed as a generalized Chi-square [5], and neither its PDF nor CDF has a closed form in the general case. These quadratic forms of gaussian vectors has been studied well and for decades [11], and some special cases does have closed forms. Some have also studied giving tail bounds on these probabilities [9], unfortunately there was none which were of use in this setting. However, there does exist several numerical algorithms for evaluating the cumulative density function with high precision [4]. I would recommend using one of these algorithms in practice, and will do so in the emperical analysis in section **??**. For the sake of analysis I will provide and upper bound on C using Bernstein's inequality.

Lemma 3.2. *Let $X_1, X_2, \ldots, X_d$ be d independent random gaussian variables where for $1 \leq j \leq d$ we have that $X_j \sim \mathcal{N}(0, \sigma_j^2)$. Let thereafter the largest standard deviation be denoted as $\sigma_* := \max_{j \in [d]} \sigma_j$, we can then bound the probability for the sum of variables squared as follows*

$$\Pr\left[\sum_{j \in [d]} X_j^2 \geq t \sqrt{8 \sum_{j \in [d]} \sigma_j^4} + \sum_{j \in [d]} \sigma_j^2\right] < e^{-t^2}$$

for

$$0 \leq t \leq \frac{1}{6\sigma_*^2} \sqrt{2 \sum_{j \in [d]} \sigma_j^4}$$

Proof

At first we define the random variable $Y_j = X_j^2 - \mathbb{E}\left[X_j^2\right]$ using the j'th gaussian random variable. As $\mathbb{E}\left[Y_j\right] = \mathbb{E}\left[X_j^2 - \mathbb{E}\left[X_j^2\right]\right] = \mathbb{E}\left[X_j^2\right] - \mathbb{E}\left[X_j^2\right] = 0$ we have that Y_j is zero centered. We are thus interested in giving bounds on $\Pr\left[\sum_{j \in [d]} Y_j\right]$. We can use Bernsteins inequality, if there exists some $L \in \mathbb{R}$ which satisfies

$$\mathbb{E}\left[|Y_j^k|\right] \leq \frac{1}{2}\mathbb{E}\left[Y_j^2\right] L^{k-2} k! \tag{13}$$

for all $k \in \mathbb{N}$ with $k \geq 2$ and for all $j \in [d]$. [14]

Initially we have that $\mathbb{E}\left[|Y_j^k|\right] = \mathbb{E}\left[\left|\left(X_j^2 - \mathbb{E}\left[X_j^2\right]\right)^k\right|\right]$ and since $X_j^2 \geq 0$ and therefore also $\mathbb{E}\left[X_j^2\right] \geq 0$ we can therefore bound it by

$$\mathbb{E}\left[\left|\left(X_j^2 - \mathbb{E}\left[X_j^2\right]\right)^k\right|\right] \leq \mathbb{E}\left[|X_j^{2k}|\right] = \mathbb{E}\left[\left|\left(\sigma_j^2 Z\right)^k\right|\right] = \sigma_j^{2k} \cdot \mathbb{E}\left[|Z^k|\right]$$

Where $Z \sim \chi_1^2$ is a chi square with 1 degree of freedom. The moment generating function of Z is $\mathbb{E}\left[Z^m\right] = 1 \cdot 3 \cdot 5 \cdot \ldots \cdot (2m - 1)$. Using this we can get the following bound

$$\sigma_j^{2k} \cdot \mathbb{E}\left[|Z^k|\right] = \sigma_j^{2k} \cdot \prod_{c=1}^k (2c - 1) = \sigma_j^{2k} \cdot 3 \cdot \prod_{c=3}^k (2c - 1)$$

$$\leq \sigma_j^{2k} \cdot 3 \cdot \prod_{c=3}^k 2c = \frac{3}{2}\sigma_j^{2k} \cdot 2^{k-2} \cdot k!$$

10

Concluding that $\mathbb{E}\left[|Y_j^k|\right] \leq \frac{3}{2}\sigma_j^{2k} \cdot 2^{k-2} \cdot k!$.

Secondly I calculate $\mathbb{E}\left[Y_j^2\right]$ exactly to be used in equation 13

$$\mathbb{E}\left[Y_j^2\right] = \mathbb{E}\left[\left(X_j^2 - \mathbb{E}\left[X_j^2\right]\right)^2\right] \tag{14}$$

$$= \mathbb{E}\left[X_j^4 + \mathbb{E}\left[X_j^2\right]^2 - 2X_j^2\mathbb{E}\left[X_j^2\right]\right] \tag{15}$$

$$= \mathbb{E}\left[X_j^4\right] + \mathbb{E}\left[X_j^2\right]^2 - 2\mathbb{E}\left[X_j^2\right]^2 \tag{16}$$

$$= \mathbb{E}\left[X_j^4\right] - \mathbb{E}\left[X_j^2\right]^2 = \sigma_j^4\mathbb{E}\left[Z^2\right] - \left(\sigma_j^2\mathbb{E}\left[Z\right]\right)^2 \tag{17}$$

$$= 3\sigma_j^4 - \sigma_j^4 = 2\sigma_j^4 \tag{18}$$

Equation (13) is therefore rewritten to

$$\frac{3}{2}\sigma_j^{2k} \cdot 2^{k-2} \cdot k! \leq \frac{1}{2} \cdot 2\sigma_j^4 L^{k-2}k! \iff \frac{3}{2}\sigma_j^{2k-4} \cdot 2^{k-2} \leq L^{k-2}$$

This expression is then split into two cases, when $k = 2$, in which it can be seen from equation (13) that this holds for any $L \in \mathbb{R}$. The other case is $k > 2$, where we have that

$$\left(\frac{3}{2}\right)^{\frac{1}{k-2}} \sigma_j^2 \cdot 2 \leq L$$

As $\lim_{k\to\infty} \left(\frac{3}{2}\right)^{\frac{1}{k-2}} = 1$, and these constraints must hold for all $j \in [d]$, we can define $\sigma_* = \max_{j\in[d]} \sigma_j$ and finally have that

$$L = 3\sigma_*^2$$

We can then use Bernstein's inequality to bound the following

$$\Pr\left[\sum_{j\in[d]} Y_j \geq 2t\sqrt{\sum_{j\in[d]} \mathbb{E}\left[Y_j^2\right]}\right] < e^{-t^2} \tag{19}$$

Which in in this case can be rewritten to get a tail bound on $\sum_{j\in[d]} X_j^2$, by reusing results from equations (14)-(18)

$$\Pr\left[\sum_{j\in[d]} Y_j \geq 2t\sqrt{\sum_{j\in[d]} \mathbb{E}\left[Y_j^2\right]}\right] = \Pr\left[\sum_{j\in[d]} \left(X_j^2 - \mathbb{E}\left[X_j^2\right]\right) \geq 2t\sqrt{\sum_{j\in[d]} 2\sigma_j^4}\right]$$

$$= \Pr\left[\sum_{j\in[d]} X_j^2 \geq 2t\sqrt{\sum_{j\in[d]} 2\sigma_j^4} + \sum_{j\in[d]} \mathbb{E}\left[X_j^2\right]\right] = \Pr\left[\sum_{j\in[d]} X_j^2 \geq 2t\sqrt{\sum_{j\in[d]} 2\sigma_j^4} + \sum_{j\in[d]} \sigma_j^2\right]$$

$$= \Pr\left[\sum_{j\in[d]} X_j^2 \geq t\sqrt{8\sum_{j\in[d]} \sigma_j^4} + \sum_{j\in[d]} \sigma_j^2\right]$$

Finally giving us that

$$\Pr\left[\sum_{j\in[d]} X_j^2 \geq t\sqrt{8\sum_{j\in[d]}\sigma_j^4} + \sum_{j\in[d]}\sigma_j^2\right] < e^{-t^2}$$

as long as t lies within the following bounds

$$0 \leq t \leq \frac{1}{2L}\sqrt{\sum_{j\in[d]}\mathbb{E}\left[Y_j^2\right]} = \frac{1}{6\sigma_*^2}\sqrt{2\sum_{j\in[d]}\sigma_j^4}$$

∎

Combining lemma 3.2 with lemma 3.1 we have can conclude the following:

Theorem 5. *Let $X \in \mathbb{R}^{n\times d}$ be a dataset in which all dimensions are independently gaussian random variables, i.e. $x_{i,j} \sim \mathcal{N}(0,\sigma_j^2)$. Let $\sigma_* := \max_{j\in[d]}\sigma_j$ be the maximal standard deviation of all dimensions. We then have that when $\ln(n) \leq \frac{\sum_{j\in[d]}\sigma_j^2}{18\sigma_*^2}$ the expected error of algorithm 3 can be upper bounded by*

$$\mathbb{E}\left[\|\boldsymbol{\eta}\|^2\right] \leq 4\sigma_{opt}^2 \cdot \sum_{j\in[d]}\sigma_j \cdot \left(\sqrt{8\ln(n)\sum_{i\in[d]}\sigma_i^2} + \sum_{j\in[d]}\sigma_j\right)$$

when each b_j is chosen according to lemma 3.1.

Proof
By equation (9) the error is given by

$$\mathbb{E}\left[\|\boldsymbol{\eta}\|^2\right] = (2C\sigma_{opt})^2 \cdot \sum_{i=1}^{d} b_j^{-2} \tag{20}$$

We thus need to determine an upper bound for C, we begin by looking at the transformed data

$$\widehat{x}_i := x_i \odot \boldsymbol{b} \text{ , giving } \widehat{x}_{i,j} \sim \mathcal{N}\left(0, (b_j\sigma_j)^2\right)$$

$$\Pr\left[\|\widehat{x}_i\| \geq C\right] = \Pr\left[\|\widehat{x}_i\|^2 \geq C^2\right] = \Pr\left[\sum_{j\in[d]}\widehat{x}_{i,j}^2 \geq C^2\right]$$

Which means I can give an upper bound on C^2, whilst having $b_j\sigma_j = \sqrt{\frac{\sigma_j}{\sum_{i\in[d]}\sigma_i}}$ as the standard deviation. Furthermore, I wish the clipping probability to be less than n^{-1}, giving me

$$e^{-t^2} = \frac{1}{n} \implies t = \sqrt{\ln(n)}$$

Combining these results and using the upper bound on C from lemma 3.2 I get the following

$$C^2 \leq 2\sqrt{\ln(n)}\sqrt{2\sum_{j\in[d]}\left((b_j\sigma_j)^4\right)} + \sum_{j\in[d]}(b_j\sigma_j)^2$$

$$= \sqrt{8\ln(n)\sum_{j\in[d]}\left(\frac{\sigma_j}{\sum_{i\in[d]}\sigma_i}\right)^2} + \sum_{j\in[d]}\frac{\sigma_j}{\sum_{i\in[d]}\sigma_i}$$

$$= \frac{\sqrt{8\ln(n)\sum_{j\in[d]}\sigma_j^2}}{\sum_{i\in[d]}\sigma_i} + 1$$

Inserting this back into equation 20 we conclude

$$\mathbb{E}\left[\|\boldsymbol{\eta}\|^2\right] = 4\sigma_{opt}^2 \cdot C^2 \cdot \sum_{i=1}^{d}b_j^{-2}$$

$$\leq 4\sigma_{opt}^2 \cdot \left(\frac{\sqrt{8\ln(n)\sum_{j\in[d]}\sigma_j^2}}{\sum_{i\in[d]}\sigma_i} + 1\right) \cdot \sum_{i=1}^{d}\left(\sigma_i \cdot \sum_{j\in[d]}\sigma_j\right)$$

$$= 4\sigma_{opt}^2 \cdot \left(\frac{\sqrt{8\ln(n)\sum_{j\in[d]}\sigma_j^2}}{\sum_{i\in[d]}\sigma_i} + 1\right) \cdot \left(\sum_{i=1}^{d}\sigma_i\right)^2$$

$$= 4\sigma_{opt}^2 \cdot \sum_{i=1}^{d}\sigma_i \cdot \left(\sqrt{8\ln(n)\sum_{j\in[d]}\sigma_j^2} + \sum_{i=1}^{d}\sigma_i\right)$$

And the constraint on $t = \sqrt{\ln(n)}$ from lemma 3.2 is

$$0 \leq t \leq \frac{1}{6(b_j\sigma_*)^2}\sqrt{2\sum_{j\in[d]}(b_j\sigma_j)^4} \iff$$

$$\sqrt{\ln(n)} \leq \frac{1}{6}\frac{\sum_{j\in[d]}\sigma_j}{\sigma_*}\sqrt{\sum_{j\in[d]}\left(\frac{\sigma_j}{\sum_{i\in[d]}\sigma_i}\right)^2} \iff$$

$$\sqrt{\ln(n)} \leq \sqrt{\left(\frac{\sum_{j\in[d]}\sigma_j}{\sigma_*}\right)^2 \cdot \frac{2}{36} \cdot \sum_{j\in[d]}\left(\frac{\sigma_j}{\sum_{i\in[d]}\sigma_i}\right)^2} \iff$$

$$\ln(n) \leq \frac{1}{18\sigma_*^2} \cdot \sum_{j\in[d]}\sigma_j^2 \iff$$

$$\ln(n) \leq \frac{\sum_{j\in[d]}\sigma_j^2}{18\sigma_*^2}$$

$\blacksquare$

At this point it would be preferable to compare the error of algorithm 3, to that of doing no transformation, and similairly deciding on a bound C such that the clipping

probability was less than n^{-1} on X. However as I can only give upper bounds on these errors there is little to no value in comparing upper bounds as there is no guarentee on the thighness of these bounds. So even if I were to determine an upper bound for the error without transformation which was higher than that of algorithm 3, it could very well still be that the error was less in reality. As previously mention, there exists numerical solutions for evaluating $\Pr\left[\|x_i\| > C^2\right]$ so it is natural to perform an emperically evaluation of these errors and investigating whether the transformation lessen the error. Intuitively it makes sense that allowing different noise to be added for each dimension, can only produce better results.

> Write and discuss restrictiveness of the bound

> Write about alternative ways of giving upper bound on C, and why it is not so important

> Write alternative ways of defining the problem. CLip prob less than n^{-1} and such

Earlier when I defined the problem at hand I also defined the constraint that the expected norm after transforming the data should be 1, i.e. the constraint given by equation (10). This constraint is perhaps not optimal in terms of how the problem should be restricted. The idea behind the constraint is to force the data to become more symmetrical in the transformed space, however it does not necessarily enforce this. An alternative constraint which I initially investigated for this project was to find a transformation, such that the majority of data entries lie within a ball with diameter 1, and entries outside would be clipped. In this transformed space it would be sufficient to add noise drawn from $\mathcal{N}(0, \sigma_{opt}^2)$. The constraint would thus be the following

$$\Pr\left[\|x_i \odot \boldsymbol{b}\| > \frac{1}{2}\right] \leq \frac{1}{n} \tag{21}$$

To achieve optimal results with this constraint it would be required to know the cumulative density function of $\|x_i \odot \boldsymbol{b}\|$, which as discussed earlier is distributed as a generalized chi square and does not have a closed form. So without a closed form there would be no way of finding optimal values for b_j. Another option would be to have the same constraint, but instead define it in terms of a tail inequality such as the one given by lemma 3.2. However the constraint on t in that lemma can be quite restrictive for some inputs, and the method would not be applicable to all datasets. I also attempted to find optimal values for b_j whilst using Chebyshev's inequality to give closed forms for equation (21), unfortunately this lead to a very complicated optimization problem, which I did not attempt to solve. The benefit os using Chebyshev's inequality would be that it is very generalized and would be applicable to any dataset, but it is also quite well known that the bounds given by Chebyshev's inequality are usually quite loose. I settled for the constraint where the expected norm is 1, which intuitively seemed quite good and was much easier to solve. In the next section I will investigate the benefit of transforming the data in an emperical setting.

> Write emperical evaluation section

4 Emperical Evaluation

For the emperical evaluation I wanted to investigate the improvement by calculating the expected error on hypothetical datasets. This is as a contrast to using real world datasets, where it is uncertain whether the data is distributed according to a multivariate gaussian mechanism with independent dimensions. Secondly I gain the benefit of creating a wide range of hypothetical scenarios and evaluating these. When creating hypothetical dataset I am merely interested in the standard deviations of each dimension, thus there is no need to truly create a dataset but instead I chose to synthetically create the parameters of a hypothetical dataset. So I chose to have the standard deviation of each dimension be chosen according to Zipf's law [13]. The distribution is often used to describe sizes of cities or frequencies of words in texts, but it can also be used to describe discrete distributions with different levels of skewness. This propoerty is what I will be utilizing, as I will chose the standard deviation of dimension $i \in [d]$ as

$$\sigma_i = \frac{\frac{1}{i^\alpha}}{\sum_{j \in [d]} \frac{1}{j^\alpha}}$$

The parameter α thus decides the skewness of how much each dimension contributes to the total amount of variance. Figure 4 shows how much each dimension contributes to the total variance for $\alpha \in \{0.1, 1, 3\}$ and $d = 100$. This means that for high α only a few dimensions has any variance, and for low α it is almost uniformly distributed.
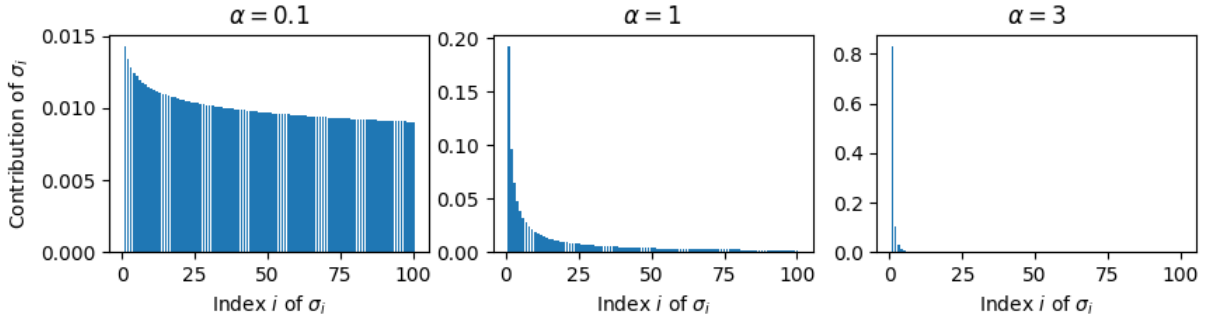

Figure 1: Examples of how standard deviations are distributed according to zipf's law for different skewness levels (α).

So the needed parameters to describe a hypothetical are α, d the number of dimensions, and n the number of points which is needed to decide the clipping probability.

I then calculated the expected error as described by equation (20) denoting this as $\mathcal{E}_t$ and compare that to

$$\mathcal{E}_n = (2C_n \sigma_{opt})^2$$

In which C_n is choose as the minimal value satisfying $\Pr\left[\|x_i\|^2 > C_n^2\right] \leq \frac{1}{n}$. I then calculate the fraction $\frac{\mathcal{E}_n}{\mathcal{E}_t}$ to compare the expected errors, and as I have σ_{opt}^2 in both the denominator and the numerator I can ommit caluclating this as doing so is not a trivial task. Figure 4 shows calculating this fraction for varying $\alpha \in [10^{-2}, 10^2]$, and for $d \in \{10, 100, 1000\}$ and $n \in \{10^2, 10^3, 10^6\}$. It shows that when few dimensions have much more variance than

15

the rest, performing the transformation improes the expected error, for all configurations of n and d.
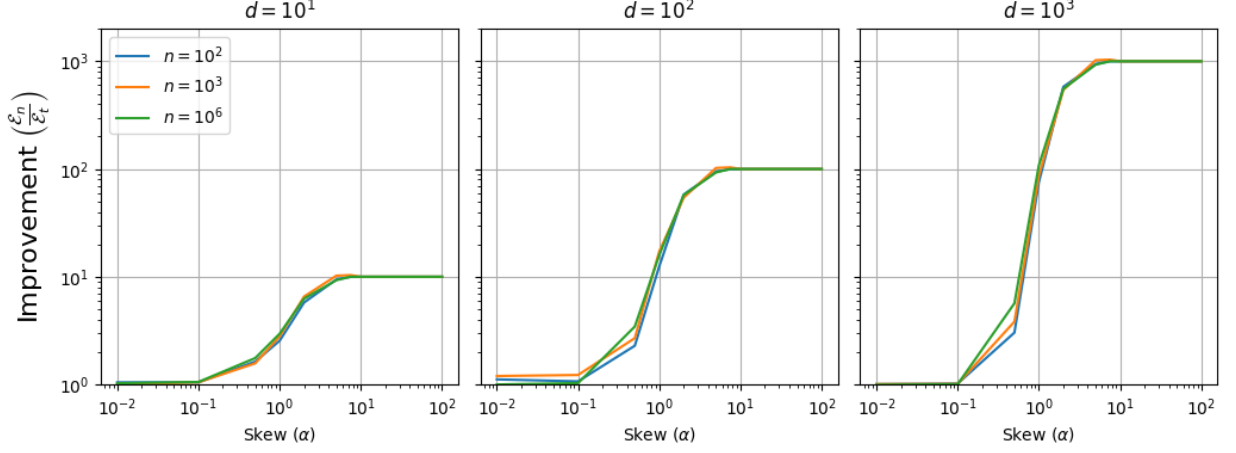

Figure 2: Improvement of applying the transformation. Each subplot shows for a fixed number of dimensions d, how much the improvement is for different skewness paramters α and number of data points n, which is used to decide the clipping probability.

More generally it seems as though n has no impact on the improvement of the method, which is not trivial as the clipping probability is chosen as $\frac{1}{n}$. It could perhaps have been that the transformation changed the distribution in such a way that the tail was much longer. Interestingly I observe that the method improves by a factor of $[1, d]$ quite like how the Elliptical Gaussian Mechanism improves the Gaussian Mechanism by a factor of $[1, d]$. Perhaps a similiar argument could be made in this case, if the distribution of the norms had a closed form or more analysis of it was available. But the emperical evidence is quite strong in favor of this phenomenon being present here as well.

In this setting To create these hypothetical dataset, I

in expected error by calculating

Downside of not creating dataset adn doing algo is that clipping bias is not taken into account

Compare method to state of the art on hypothetical datasets.

Mention that method requires sigmas to be known as estimating would take up most of the privacy budget.

16

5 It is a mess, don't look here

5.1 Gaussian data

Let $X^{(j)} \sim \mathcal{N}(0, \sigma_j^2)$ As the expected l_2-norm of x_i is given by

$$\mathbb{E}\left[\|x_i\|^2\right] = \sum_{j=1}^d \sigma_j^2$$

To achieve an expected norm of 1 I will scale each dimension by a factor $\frac{1}{b_j}$ which achieves this. If

$$\hat{x}_i = \left(\frac{x_i^{(0)}}{b_0}, \frac{x_i^{(1)}}{b_1}, \ldots, \frac{x_i^{(d)}}{b_d}\right)$$

This means that $X^{(j)} \sim \mathcal{N}(0, \frac{\sigma_j^2}{b_j^2})$ and the expected norm is given by

$$\mathbb{E}\left[\|x_i\|^2\right] = \sum_{j=1}^d \frac{\sigma_j^2}{b_j^2}$$

and I can introduce that constraint that the expected norm after the transformation should be 1. In such a case when noise is added after the transformation $\hat{X} + \eta$ where $\eta \sim N(0, t^2)$ and achieves (ε, δ)-DP in this space then then due to linearity of transformation the noise introduced in the original space is then given by then the error is

I desire a transformation of $x_i^{(j)}$ such that the expected norm is 1. Thus I must scale each dimension by $\frac{1}{b_j}$, and have that

$$\mathbb{E}\left[\|x_i\|^2\right] = 1$$

Minimize $\|\hat{\eta}\|$ under the constraint that $\mathbb{E}\left[\|x_i\|^2\right] = 1$

Lemma 5.1. *Let $X \sim \mathcal{N}(0, \sigma^2)$, and Φ denote the cumulative density function of $\mathcal{N}(0, 1)$, then the cumulative density function of X^2 is given by*

$$F_{X^2}(x) = \Pr\left[X^2 \leq x\right] = 2\Phi\left(\frac{\sqrt{x}}{\sigma}\right) - 1$$

Proof

$$\Pr\left[X^2 \leq x\right] = \Pr\left[|X| \leq \sqrt{x}\right] = 2\Pr\left[0 \leq X \leq \sqrt{x}\right]$$

$$= 2\left(\Pr\left[X \leq \sqrt{x}\right] - \Pr\left[X \leq 0\right]\right) = 2\left(\Pr\left[X \leq \sqrt{x}\right] - \frac{1}{2}\right)$$

$$= 2\Phi\left(\frac{\sqrt{x}}{\sigma}\right) - 1$$

∎

Corollary 5.1. *From lemma 5.1 we can give following bound for $X \sim \mathcal{N}(0, \sigma^2)$.*

$$\Pr\left[X^2 > (4\sigma)^2\right] < 10^{-4}$$

Bernsteins inequality then states that

$$\Pr\left[\sum_{j\in[d]} X_j \geq 2t\sqrt{\sum_{j\in[d]} \mathbb{E}\left[X_i^2\right]}\right] < e^{-t^2}$$

for all

$$0 \leq t \leq \frac{1}{2L}\sqrt{\sum}$$

I have that $\bar{X}$ is a standard gaussian variable.
Let $Y_j = X_j^2$ I am interested in $\sum_{j\in[d]} Y_j$, then it has

$$\mathbb{E}\left[|Y_j^k|\right] = \mathbb{E}\left[|X_j^{2k}|\right] = \mathbb{E}\left[|X_j|^{2k}\right] = \mathbb{E}\left[|\sigma_j \bar{X}|^{2k}\right] =$$

$$|\sigma_j|^{2k}\mathbb{E}\left[|\bar{X}|^{2k}\right] = \sigma_j^{2k}\prod_{c=1}^{k} 2c = \sigma_j^{2k} \cdot 2^k \cdot k!$$

Which means $\mathbb{E}\left[Y_j^2\right] = 8\sigma_j^4$. To check the constraint (find value for L)

$$\sigma_j^{2k} \cdot 2^k \cdot k! \leq \frac{1}{2} \cdot 8\sigma_j^4 \cdot L^{k-2}k! \iff$$

$$\sigma_j^{2k} \cdot 2^k \leq 4\sigma_j^4 \cdot L^{k-2} \iff$$

$$\sigma_j^{2(k-2)} \cdot 2^k \leq 4\mathrm{L}^{k-2}$$

We have that $\sigma_j = \sigma_j b_j = \frac{\sqrt{\sigma_j}}{\sqrt{\sum_{i\in[d]}\sigma_i}} = \sqrt{\frac{\sigma_j}{\sum_{i\in[d]}\sigma_i}}$. Therefore

$$\sigma_j^{2(k-2)} \cdot 2^k = \left(\sqrt{\frac{\sigma_j}{\sum_{i\in[d]}\sigma_i}}\right)^{2(k-2)} \cdot 2^k =$$

$$\left(\frac{\sigma_j}{\sum_{i\in[d]}\sigma_i}\right)^{k-2} \cdot 2^k \leq 2^k$$

This therefore implies

$$2^k \leq 4L^{k-2} \iff 2^{k-2} \leq L^{k-2} \iff 2 \leq L$$

If it is desired that less than 10^{-p} points are removed then $t = \sqrt{p\ln(10)}$ as long as

$$\sqrt{p\ln(10)} \leq \frac{1}{2L}\sqrt{\sum_{j\in[d]}\mathbb{E}\left[X_j^2\right]} = \sqrt{4} \cdot \frac{\sqrt{\sum_{i\in[d]}\sigma_i^2}}{\sum_{i\in[d]}\sigma_i}$$

An alternative Again minimize $\|\hat{\eta}\|$, but instead the constraint comes from Chebyshev's inequality, I always have that

$$\Pr\left[\|\|x_i\|^2 - \mathbb{E}\left[\|x_i\|^2\right]\| \geq k \cdot \sqrt{\mathrm{Var}\left[\|x_i\|^2\right]}\right] \leq \frac{1}{k^2}$$

18

Therefore I can set $\frac{1}{k^2} = 0.05$, and find a transformation where I decide how many standard deviations I must be away from the mean to have norm greater than 1, i.e.

$$\mathbb{E}\left[\|x_i\|^2\right] + k \cdot \sqrt{\mathrm{Var}\left[\|x_i\|^2\right]} = 1 \implies k = \frac{1 - \mathbb{E}\left[\|x_i\|^2\right]}{\sqrt{\mathrm{Var}\left[\|x_i\|^2\right]}}$$

I then have that

$$\frac{1}{k^2} = \frac{1}{\left(\frac{1-\mathbb{E}[\|x_i\|^2]}{\sqrt{\mathrm{Var}[\|x_i\|^2]}}\right)^2} = \left(\frac{\sqrt{\mathrm{Var}\left[\|x_i\|^2\right]}}{1 - \mathbb{E}\left[\|x_i\|^2\right]}\right)^2 = \frac{\mathrm{Var}\left[\|x_i\|^2\right]}{(1 - \mathbb{E}\left[\|x_i\|^2\right])^2}$$

I already know that

$$\mathbb{E}\left[\|x_i\|^2\right] = \sum_{j=1}^{d} \frac{\sigma_j^2}{b_j^2}$$

$$\mathrm{Var}\left[\|x_i\|^2\right] = 2\sum_{j=1}^{d} \frac{\sigma_j^4}{b_j^4}$$

I therefore have the constraint

$$\frac{\mathrm{Var}\left[\|x_i\|^2\right]}{(1 - \mathbb{E}\left[\|x_i\|^2\right])^2} = \frac{2\sum_{j=1}^{d}\frac{\sigma_j^4}{b_j^4}}{\left(1 - \sum_{j=1}^{d}\frac{\sigma_j^2}{b_j^2}\right)^2} = 0.05$$

Be aware that this could find cases where expected norm is greater than 1 and the constraint then says that they are never less than 1 in norm. Another restriction could be that expected norm is ¡ 1.

Chebyshev's inequality could also be used when expected norm should be 1 and then put a bound on number of std away one must be to have less than 0.05 fraction of data removed.

Extra

Determining α using Bernsteins inequality

$$\Pr\left[\|x_i\|^2 - \mathbb{E}\left[\|x_i\|^2\right] > t\right] < 2 \cdot \exp\left(-\frac{t^2/2}{\mathrm{Var}\left[\|x_i\|^2\right] + C \cdot t/3}\right)$$

In the case where $\mathbb{E}\left[\|x_i\|^2\right] = 1$ and b_i is optimized in this case, we have that

$$P(\|x_i\|^2 > 1 + t) < 2\exp\left(-\frac{t^2/2}{\mathrm{Var}\left[\|x_i\|^2\right] + C \cdot t/3}\right)$$

$$\mathrm{Var}\left[\|x_i\|^2\right] = 2\frac{\sum_i^d \sigma_i^2}{\left(\sum_i^d \sigma_i\right)^2}$$

$$C = \max_{i \in [d]}\left(\sigma_i\right) \cdot \frac{16}{\sum_i^d \sigma_i}$$

C is decided such that less than 0.0001 fraction of the data is outside this bound in each dimension. i.e.

$$\forall j \in [d] : \ \Pr\left[X^{(j)} > C\right] < 0.0001$$

Solving for t

$$2 \cdot \exp\left(-\frac{t^2/2}{\mathrm{Var}\left[\|x_i\|^2\right] + C \cdot t/3}\right) = 0.0001 \iff \ln\left(\frac{0.0001}{2}\right) = -\frac{t^2/2}{\mathrm{Var}\left[\|x_i\|^2\right] + C \cdot t/3} \implies$$

$$t = -\frac{C\ln\left(\frac{0.0001}{2}\right) - \sqrt{\ln\left(\frac{0.0001}{2}\right)\left(-18 \cdot \mathrm{Var}\left[\|x_i\|^2\right] + C^2\ln\left(\frac{0.0001}{2}\right)\right)}}{3}$$

Then we have that $\alpha = 1 + t$.

Determine α for the bound using `https://en.wikipedia.org/wiki/Concentration_inequality`, or `https://web.stanford.edu/class/cs229t/2017/Lectures/concentration-slides.pdf`

References

[1] BALLE, B., AND WANG, Y. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. *CoRR abs/1805.06530* (2018).

[2] BATES, D. *Quadratic Forms of Random Variables.* University of Wisconsin-Madison: STAT 849 lectures, 2010.

[3] BISWAS, S., DONG, Y., KAMATH, G., AND ULLMAN, J. Coinpress: Practical private mean and covariance estimation. *Advances in Neural Information Processing Systems 33* (2020), 14475–14485.

[4] CHEN, T., AND LUMLEY, T. Numerical evaluation of methods approximating the distribution of a large quadratic form in normal variables. *Computational Statistics & Data Analysis 139* (2019), 75–81.

[5] DAS, A., AND GEISLER, W. S. A method to integrate and classify normal distributions, 2020.

[6] DWORK, C., MCSHERRY, F., NISSIM, K., AND SMITH, A. Calibrating noise to sensitivity in private data analysis. *Journal of Privacy and Confidentiality 7*, 3 (2016), 17–51.

[7] DWORK, C., ROTH, A., ET AL. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science 9*, 3–4 (2014), 211–407.

[8] HALLAM, A. *Some Theorems on Quadratic Forms and Normal Variables.* Iowa State University: Econ 671 lectures, 2004.

[9] HANSON, D. L., AND WRIGHT, F. T. A bound on tail probabilities for quadratic forms in independent random variables. *The Annals of Mathematical Statistics 42*, 3 (1971), 1079–1083.

[10] HUANG, Z., LIANG, Y., AND YI, K. Instance-optimal mean estimation under differential privacy.

[11] MATHAI, A., AND PROVOST, S. Quadratic forms in random variables.

[12] PAGH, R., AND LEBEDA, C. Private vector aggregation when coordinates have different sensitivity.

[13] POWERS, D. M. W. Applications and explanations of Zipf's law. In *New Methods in Language Processing and Computational Natural Language Learning* (1998).

[14] WAINWRIGHT, M. J. *Basic tail and concentration bounds.* Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019, p. 21–57.