Statistic Selection and MCMC for Differentially Private Bayesian Estimation

Barış Alparslan and Sinan Yıldırım

Faculty of Engineering and Natural Sciences, Sabancı University, İstanbul, Turkey

baris.alparslan@sabanciuniv.edu, sinanyildirim@sabanciuniv.edu

March 29, 2022

Abstract

This paper concerns differentially private Bayesian estimation of the parameters of a population distribution, when a statistic of a sample from that population is shared in noise to provide differential privacy.

This work mainly addresses two problems: (1) What statistic of the sample should be shared privately? For the first question, i.e., the one about statistic selection, we promote using the Fisher information. We find out that, the statistic that is most informative in a non-privacy setting may not be the optimal choice under the privacy restrictions. We provide several examples to support that point. We consider several types of data sharing settings and propose several Monte Carlo-based numerical estimation methods for calculating the Fisher information for those settings. The second question concerns inference: (2) Based on the shared statistics, how could we perform effective Bayesian inference? We propose several Markov chain Monte Carlo (MCMC) algorithms for sampling from the posterior distribution of the parameter given the noisy statistic. The proposed MCMC algorithms can be preferred over one another depending on the problem. For example, when the shared statistics is additive and added Gaussian noise, a simple Metropolis-Hasting algorithm that utilizes the central limit theorem is a decent choice. We propose more advanced MCMC algorithms for several other cases of practical relevance.

Our numerical examples involve comparing several candidate statistics to be shared privately. For each statistic, we perform Bayesian estimation based on the posterior distribution conditional on the privatized version of that statistic. We demonstrate that, the relative performance of a statistic, in terms of the mean squared error of the Bayesian estimator based on the corresponding privatized statistic, is adequately predicted by the Fisher information of the privatized statistic.

Keywords: Differential privacy; Markov chain Monte Carlo; Fisher Information; Statistic selection; Bayesian Statistics.

1 Introduction

In recent years, differential privacy has become a popular framework for achieving privacy-preserving data sharing and inferential analysis of sensitive data sets (Dwork, 2006; Dwork and Roth, 2013). In this paper, we are interested in differentially private Bayesian estimation for the parameters of a population distribution, when a statistic of a sample from that population is shared in noise so as to provide differential privacy. This work concerns two problems:

1. What statistic of the sample should be shared?

2. Based on the shared statistics, how could we make inference?

For the first question, i.e., the one about statistic selection, we consider using the Fisher information. For the second question, we propose Markov chain Monte Carlo (MCMC) in general, to draw samples from the posterior distribution of the parameter given the noisy statistic.

Bayesian inference using MCMC has been recently studied in the data privacy context. One of the first works concerning using Monte Carlo for differentially private posterior sampling is Wang et al. (2015). In Wang et al. (2015), a class of Stochastic Gradient MCMC techniques are adapted for differential privacy. The scheme is later improved by a few works including Li et al. (2019). A general purpose and scalable differentially private MCMC algorithm was proposed in Heikkilä et al. (2019). Both Wang et al. (2015) and Heikkilä et al. (2019) lead to non-exact MCMC algorithms, in the sense that the target distribution is sampled from only asymptotically. Yıldırım and Ermiş (2019) developed an exact MCMC algorithm based on the penalty algorithm that targets the posterior distribution and provides differential privacy at the same time. The penalty algorithm is based on the classical Metropolis-Hastings algorithm; the adoption of the differentially private penalty algorithm in Yıldırım and Ermiş (2019) for Hamiltonian Monte Carlo is recently proposed in Räisä et al. (2021).

The above-mentioned works propose MCMC algorithms where in every iteration some function of the sensitive data is revealed in noise so that the iterations are made differentially private. Consider, for example, two parties, an analyst and a data-holder, where the analyst wishes to perform inference based on the sensitive data held secret by the data-holder. By their nature, the algorithms mentioned above require ongoing queries to the database. Although this can be available in some cases, it may not be practical in other situations due to the requirement of continuous interaction between the two parties as long as the course of the algorithm. Instead, in such situations it may be more feasible for the data-holder to share the data in a private manner once and for all and for the analyst to perform inference based on the shared statistic without further interaction with the data-holder. In this paper we consider the latter case, i.e., the one where instead of the involved interaction between iterations, summaries of data are shared privately prior to statistical analysis.

Foulds et al. (2016) considered adding Gaussian noise to the statistics and showed the asymptotic properties of posterior distribution when the noisy statistics are used as if the true values. The differential privacy of the generic Metropolis-Hastings algorithm is also analyzed in Foulds et al. (2016). Foulds et al. (2016) then proposed Gibbs sampling

for problems when the likelihood belongs to an exponential family. The restriction to exponential families can be indeed limiting. In theory, with advanced MCMC methodology, one can sample from the posterior distribution of a parameter when any informative statistic of the sensitive data is shared. Moreover, the method of Foulds et al. (2016) is only asymptotically biased as it does not account for the added noise to the sufficient statistics in its model.

Unlike Foulds et al. (2016), the works of Williams and Mcsherry (2010); Karwa et al. (2014); Bernstein and Sheldon (2018); Gong (2019) correctly accommodate the shared noisy statistic of the sensitive data into a hierarchical model which has the structure

$$\text{parameter} \rightarrow \text{sensitive data} \rightarrow \text{noisy statistic}.$$

The posterior distribution based on the noisy statistic has very strong resemblance to the already existing approximate Bayesian computation (ABC) literature. Although the hierarchical model above has been considered earlier, e.g. in Williams and Mcsherry (2010); Karwa et al. (2014); Bernstein and Sheldon (2018), the relation between differentially private statistics and approximate Bayesian computation, in particular noisy ABC, is pronounced for the first time in Gong (2019). While Bernstein and Sheldon (2018) proposed Gibbs sampler for the hierarchical model and released samples from the posterior using two stage updating process, a rejection sampler for the ABC posterior as well as an expectation-maximization (EM) algorithm is proposed in Gong (2019). In a following work, Park et al. (2021) developed an differentially private ABC method in maximum mean discrepancy is used as the distance metric between artificial and observed data and the acceptance probability is randomized.

In ABC, the artificial and real data are usually compared via a statistic. In this paper, we consider the following scenario. A statistic of the sensitive data is shared in a privacy preserving manner. Then, one samples from the noisy ABC posterior of the parameter of interest conditional on the noisy statistic. Nevertheless, the choice of the statistic is important for inference from finite data: In the case without privacy concerns, one would like to choose the statistic that is most informative about the parameter to be estimated (Fearnhead and Prangle, 2012). When ABC is done in a DP context, however, the most informative statistic in the non-private setting is not necessarily the best choice. This is because the statistic is revealed in privacy preserving noise and the noise variance depends on the sensitivity of the statistic. In order to determine the best choice for the statistic to be shared, one must compare among the informativeness of the *noisy* statistics.

The question of scope and efficiency of statistical learning with differential privacy has been studied in the literature (Kasiviswanathan et al., 2008; Dwork and Lei, 2009; Dwork and Smith, 2010; Smith, 2011; Lei, 2011). Kasiviswanathan et al. (2008) demonstrated that, in most problems where the relation between examples to labels are learned, a private learner can learn what a non-private learner can in the same order of the number of samples. Smith (2011) established the existence of differentially private estimators with the same asymptotic variance as their non-private counterparts, also proposing such an estimator which can be seen as an improved version of those in Dwork and Lei (2009); Dwork and Smith (2010). Both Smith (2011) and Dwork and Lei (2009); Dwork and Smith (2010) are based on the subsample and aggregate technique of Nissim et al. (2007). There

also exist related works where robust statistics and M-estimators (Lei, 2011; Smith, 2011; Avella-Medina, 2019) are studied in a data privacy context.

Although the works mentioned above contain methods with certain convergence guarantees, they are not directly concerned with choosing the best representation of the data, either individually or in an aggregate fashion. This motivates our first contribution of the paper, which is a method for statistic selection to be used in the private data-sharing step. We propose to use the Fisher information contained in the noisy statistic for the parameter as the criterion to compare. The Fisher information is a relevant measure when one wishes to use a likelihood-based estimation for the unknown parameter, such as maximum likelihood estimation and Bayesian estimation. The Fisher information is a function of the parameter and it depends jointly the statistic, its sensitivity, and the targeted privacy level.

The statistic selection step of private data sharing indeed bears practical importance. Take, for example, two choices for the shared statistic of a sensitive sample $X_{1:n}$ from a normal population with mean 0 and an unknown variance: one being the sample average of squares X_i^2 and the other being the sample average of the absolute values $|X_i|$. While the order of the sample size to learn the unknown variance is the same for both choices, a non-asymptotic analysis will reveal one of them preferable over the other. Indeed, in Examples 1, 2, and 3, we show on simple distributions that the conventional statistics may not be the best choices to share the data privately.

As a second contribution, we propose effective MCMC algorithms that target the true posterior of the noisy ABC based on the noisy statistic. Obviously, there is no ideal algorithm that performs best in all the scenarios considered here. However, the broad family of exact-approximate MCMC algorithms offer effective choices. We inspect specifically algorithms that are based on pseudo-marginal MCMC (Andrieu and Roberts, 2009) and the recently introduced framework called Metropolis-Hastings with averaged acceptance ratio (MHAAR) in (Andrieu et al., 2020).

The organization of the paper is as follows. In Section 2, we introduce the basic concepts of differential privacy. In Section 3, we discuss the problem of parameter estimation using privatized noisy statistics of the sensitive data and propose Fisher information as a measure of informativeness of the shared statistic (in noise). We show with analytical examples that, according to Fisher information, sharing non-standard statistics for a population parameter may be more beneficial compared to standard statistics. Section 4 is reserved for the MCMC based Bayesian inference algorithms proposed for the models induced by the privacy preserving sharing scenarios described in Section 3. In Section 5 we present the results of some numerical experiments. Finally, we give our concluding remarks and possible future work in Section 6.

2   Differential Privacy

In this section, we take differential privacy as the primary definition of data privacy; although we also mention other closely related definitions.

Let $\mathcal{X}$ be a universal set of individual data values. We call two data sets $x_{1:n}, x'_{1:n} \in \mathcal{X}^n$ neighbors if $x'_{1:n}$ can be obtained by changing the value of a single entry in $x_{1:n}$. In other

words, the Hamming distance between the data sets, shown as $h(x_{1:n}, x'_{1:n})$ and defined as as the number of different elements between $x_{1:n}$ and $x'_{1:n}$, is equal to 1. We call $\mathcal{A}$ a randomized algorithm whose output upon taking the input $x_{1:n}$ is a random variable $\mathcal{A}(x_{1:n})$ taking values from some $\mathcal{Y}$.

Definition 1 (Differential privacy). *We say that $\mathcal{A}$ is (ϵ, δ)-differentially private if, for any pair of neighboring data sets $x_{1:n}, x'_{1:n} \in \mathcal{X}^n$ from an input set and any subset of output values $O \subseteq \mathcal{Y}$, it satisfies the inequality* Dwork (2006)

$$\mathbb{P}\left[\mathcal{A}(x_{1:n}) \in O\right] \leq e^\epsilon \mathbb{P}\left[\mathcal{A}(x'_{1:n}) \in O\right] + \delta.$$

According to the above inequality, a randomized algorithm is differentially private if the probability distributions for the output obtained from two neighboring databases are '*similar*'. The privacy parameters (ϵ, δ) are desired to be as small as possible as far as privacy is concerned.

Assume that a privacy preserving algorithm is required to return the value of a function $\varphi : \mathcal{X} \mapsto \mathbb{R}$ evaluated at the sensitive data set $x_{1:n}$ in a private fashion. One basic way of achieving this is via the *Laplace mechanism* Dwork (2008), which relies on the *(global) sensitivity* of this function.

Definition 2 (Global sensitivity). *The L_p sensitivity of a function $\psi : \mathcal{X} \mapsto \mathbb{R}^{d_\psi}$ for $p \geq 1$ is given by*

$$\nabla_{\psi,p} = \sup_{x_{1:n}, x'_{1:n} : h(x_{1:n}, x'_{1:n}) = 1} \|\psi(x_{1:n}) - \psi(x'_{1:n})\|_p.$$

Theorem 1 (Laplace mechanism). *Let $\mathcal{A}$ be an algorithm that returns $\psi(x_{1:n}) + V$ on an input $x_{1:n} \in \mathcal{X}^n$, where $V_i \overset{\text{i.i.d.}}{\sim} \text{Laplace}(\nabla_{\psi,1}/\epsilon)$ for $i = 1, \ldots, d_\psi$. Then $\mathcal{A}$ is ϵ-DP.*

While the Laplace mechanism achieves pure differential privacy, i.e., with $\delta = 0$, another popular mechanism, called the Gaussian mechanism (Dwork and Roth, 2013) achieves differential privacy with $\delta > 0$. This mechanism adds Gaussian noise to $\psi(x_{1:n})$ where the variance of the noise is determined by the global sensitivity of $\psi(\cdot)$. The Gaussian mechanism is also a central tool according to other related definitions of differential privacy, such as the zero-concentrated differential privacy (Bun and Steinke, 2016) or the more recently introduced Gaussian differential privacy (Dong et al., 2022). For an example, the following theorem presents the privacy property of the Gaussian mechanism according to Gaussian differential privacy.

Theorem 2 (Gaussian differential privacy of the Gaussian mechanism (Dong et al., 2022)). *Gaussian mechanism that returns $\psi(x_{1:n}) + V$, where $V_i \sim \mathcal{N}(0, \nabla_{\psi,2}^2/\epsilon^2)$, for $i = 1, \ldots, d_\psi$, satisfies ϵ-Gaussian differential privacy.*

A similar result regarding the Gaussian mechanism also exists for the zero-concentrated differential privacy (Bun and Steinke, 2016). Moreover, the mentioned privacy definitions are interrelated; see Dong et al. (2022) and Bun and Steinke (2016) for the detailed relations.

One property of differential privacy relevant to our work is the post-processing property, which simply holds that the privacy loss is not increased by transforming the output through an algorithm independent of the private data given the output.

Theorem 3 (Post-processing). *Let $\mathcal{A}_1$ be an (ϵ, δ)-DP algorithm with inputs from X and outputs from $\mathcal{S}_1$, and let $\mathcal{A}_2 : \mathcal{S}_1 \mapsto \mathcal{S}$ be an algorithm that does not depend on X. Then, the algorithm $\mathcal{A} = (A_2 o \mathcal{A}_1)$ is (ϵ, δ)-DP.*

In this work, all the Bayesian inference algorithms in Section 4 act as post-processing operations.

3   Statistic selection based on Fisher information

We consider a data privacy setting where we have some sensitive data $X_1, \ldots, X_n \overset{\text{i.i.d}}{\sim} \mathcal{P}_\theta$ for some distribution $\mathcal{P}_\theta$ on $\mathcal{X}$ with parameter $\theta \in \Theta$. We assume that each X_i belongs to a distinct individual. We aim to infer θ based on the outputs of a differentially private operation on the sensitive data $X_{1:n}$. One example case is when a statistic of the data $S_n : \mathcal{X}^n \mapsto \mathbb{R}^{d_s}$, for some $d_s \geq 1$, is released in noise as

$$Y = S_n(X_{1:n}) + V, \quad V \sim \mathcal{P}_{\epsilon, S_n}, \tag{1}$$

where $\mathcal{P}_{\epsilon, S_n}$ is the distribution of the privacy preserving noise V whose parameter(s) is (are) adjusted according to S_n and ϵ. We will show two examples of $\mathcal{P}_{\epsilon, S_n}$ in Sections 3.1 and 3.2, which arise from the Gaussian and Laplace mechanisms, respectively. A common choice of S_n is an additive statistic as in

$$S_n(X_{1:n}) = \frac{1}{n} \sum_{i=1}^{n} s(X_i). \tag{2}$$

However, in this paper we will consider non-additive statistics as well.

In (1), (a statistic of) the collected data is released in batch manner. An alternative to that is when each X_i is shared privately as

$$Y_i = s(X_i) + V_i, \quad V_i \sim \mathcal{P}_{\epsilon, s}. \tag{3}$$

We will call this setting the sequential release, as opposed to the batch release in (1).

There are several other forms of differentially private data sharing, such as the exponential mechanism. In this paper we will confine the discussion to the scenarios in (1) and (3). However, as it will be clear, the proposed ideas can also be adopted for other mechanisms.

How should we choose the statistic S_n (or s)? We would like to make a choice (among several candidates) so that the resulting Y is most 'informative'. In this paper, we consider the Fisher information as the measure of the amount of 'information' that Y carries about θ. The Fisher information not only arises naturally in frequentist contexts, but it is also relevant to Bayesian estimation, especially for big data, owing to the Bernstein-von Mises theorem (Cam, 1986) Under certain regularity conditions, the posterior distribution tends to have a normal distribution with covariance determined by the Fisher information.

The Fisher information at θ is determined by the population distribution $\mathcal{P}_\theta$, the function S_n (or s), and the privacy level ϵ. To concretize the discussion, we confine the

Table 1: Model-method matching for calculating $F(\theta)$

Model	Method	Requirement
additive statistic, normal noise	Section 3.1	$\mu_s(\theta)$ and $\Sigma_s(\theta)$ are differentiable w.r.t. θ
additive statistic, non-gaussian noise	Algorithm 1	$\mu_s(\theta)$ and $\Sigma_s(\theta)$ are differentiable w.r.t. θ
non-additive statistic	Algorithm 2	$p(x\|\theta)$ is differentiable w.r.t. θ
sequential release	Algorithm 3	$p(x\|\theta)$ is differentiable w.r.t. θ

attention to the batch sharing setting in (1). The marginal density of $Y = y$ given θ can be written as

$$p_{\epsilon,S_n}(y|\theta) = \int p_{\epsilon,S_n}(y|x_{1:n}) \prod_{i=1}^{n} p(x_i|\theta) dx_{1:n}. \tag{4}$$

The Fisher information with respect to this marginal distribution can be expressed as

$$F(\theta) = \mathbb{E}\left[-\frac{\partial^2 \log p_{\epsilon,S_n}(Y|\theta)}{\partial\theta\partial\theta^T}\right] \tag{5}$$

$$= \mathbb{E}\left[\gamma_{\epsilon,S_n}(\theta;y)\gamma_{\epsilon,S_n}(\theta;y)^T\right], \tag{6}$$

where $\gamma_{\epsilon,S_n}(\theta;y)$ is the well-known score vector, defined as

$$\gamma_{\epsilon,S_n}(\theta;y) = \frac{\partial \log p_{\epsilon,S_n}(Y|\theta)}{\partial\theta}.$$

Whether $F(\theta)$ above can be calculated exactly or not and how it should be calculated approximately in the latter case depend on the nature of the statistic and/or the privacy preserving mechanism. Specifically for (5), it is critical whether the statistic is additive or not, and/or the privacy preserving noise is Gaussian or not. Furthermore, the approach to calculate $F(\theta)$ also depends on whether the data is shared in a batch or sequential manner.

Clearly, $F(\theta)$ is a function of θ and one cannot know the informativeness of the selected statistic for the stochastic process in question without knowing the true value θ that governs the process. This appears to be an issue in applying the proposed strategy of choosing statistics based on $F(\theta)$. However, the proposed strategy can be useful in several ways. For example, in some cases one statistic can be shown to yield a larger $F(\theta)$ than another uniformly over the domain of θ (see Example 2.) In other cases $F(\theta)$ can be combined with the prior distribution of θ, say $\eta(\theta)$, to come up with an overall score such as $\int F(\theta)\eta(\theta)d\theta$. Finally, when one statistic is not uniformly better in terms of $F(\theta)$ than the other statistic and no prior information is available, an initial chunk of the data can be used to obtain a posterior distribution, which is then to be used to determine the best statistic as well as to act as the prior distribution for the rest of the data.

In the rest of this section, we propose algorithms to (approximately) compute $F(\theta)$ under several practically relevant combinations of those mentioned conditions. Table 1 shows the scenario-algorithm matching which indicates the most suitable algorithm to compute $F(\theta)$ for the scenario considered.

3.1 Fisher information with additive statistic and Gaussian noise

In the classical DP setting, imperfect privacy, i.e, (ϵ, δ)-DP for $\delta > 0$, can be obtained via the Gaussian mechanism (Dwork and Roth, 2013). The Gaussian mechanism is not only a popular choice in differential privacy studies, but also *the* natural choice for Gaussian differential privacy (Dong et al., 2022), a privacy definition that leads to a more interpretable Neyman-Pearson type error analysis than the classical differential privacy.

The Gaussian mechanism is a noise adding mechanism which can be described generally as

$$Y = S_n(X_{1:n}) + V, \quad V \sim \mathcal{N}(0, \sigma_{s,n,\epsilon}^2 I). \tag{7}$$

For ease of exposition, one can take $\sigma_{s,n,\epsilon}^2 = \Delta_{s,2}^2/(n^2\epsilon^2)$, where, recall that, $\Delta_{s,2}$ is the L_2 sensitivity of $s(\cdot)$. Here, the parameter ϵ has a slightly different meaning than in the definition of classical differential privacy. Specifically, the above choice for the noise distribution provides ϵ-Gaussian DP and *not* ϵ-DP. We could make the variance also depend on a $\delta > 0$ parameter to provide (ϵ, δ)-DP, but this would distract the main messages of the discussion.

Suppose that S_n is additive as in (2). Then one can employ a normal approximation for the distribution of $S_n(X_{1:n})$, along the lines of Bernstein and Sheldon (2018). Let

$$\mu_s(\theta) = \mathbb{E}_\theta[s(X)], \quad \Sigma_s(\theta) = \text{Var}_\theta[s(X)]$$

be the mean and covariance of $s(X)$. For large n, the additive statistic approximately has a normal distribution

$$S_n(X_{1:n}) \sim \mathcal{N}(\mu_s(\theta), \Sigma_s(\theta)/n), \tag{8}$$

Combining (8) with (7), the marginal distribution of Y is approximated as

$$Y \sim \mathcal{N}\left(\mu_s(\theta), \Sigma_s(\theta)/n + \sigma_{s,n,\epsilon}^2 I\right). \tag{9}$$

Finally, considering the transformation

$$\theta \mapsto \left[\mu_s(\theta), \Sigma_s(\theta)/n + \sigma_{s,n,\epsilon}^2 I\right],$$

the (i,j)'th element of $F(\theta)$ for the distribution in (9) is given by

$$[F(\theta)]_{i,j} = \frac{\partial \mu_s(\theta)^T}{\partial \theta_i} H_{s,\epsilon,n}(\theta)^{-1} \frac{\partial \mu_s(\theta)}{\partial \theta_j} + \frac{\text{tr}(G)}{2}$$

where $H_{s,\epsilon,n}(\theta) := \frac{\Sigma_s(\theta)}{n} + \sigma_{s,n,\epsilon}^2 I$ is the covariance of Y and

$$G = \frac{1}{n^2}\left(H_{s,\epsilon,n}(\theta)^{-1} \frac{\partial \Sigma_s(\theta)}{\partial \theta_i} H_{s,\epsilon,n}(\theta)^{-1} \frac{\partial \Sigma_s(\theta)}{\partial \theta_j}\right).$$

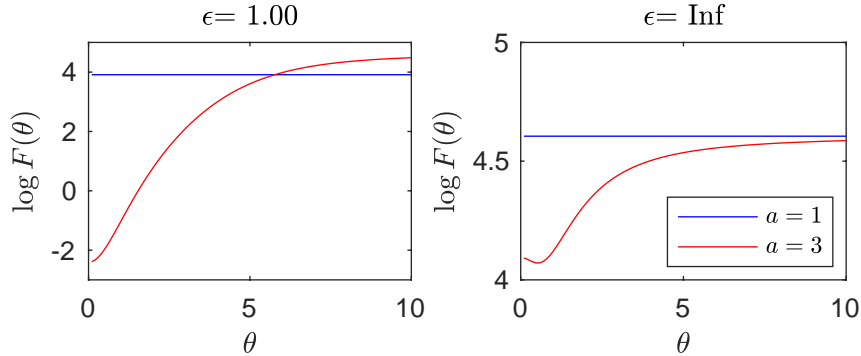Examples 1, 2, and 3 demonstrate how the proposed scheme can be used in simple but common inference problems.

Figure 1: $F(\theta)$ for the mean parameter of $\mathcal{N}(\theta, 1)$ when $s(x) = |x|^a$. Left: $\epsilon = 1$, Right: $\epsilon = \infty$ (non-private case).

Example 1 (Mean of the normal distribution). *Assume that $\mathcal{X} = (0, A)$ and the considered population distribution for X is $\mathcal{P}_\theta = \mathcal{N}(\theta, 1)$. Here A is a number which arises due to the nature of the data generation process, large enough to have negligible effect on the distribution of X (The same will be assumed in the other examples in the paper.) For statistic selection, one may want to use $s(x) = x^a$, where a is an odd integer. Let us compare $a = 1$ and $a = 3$. We have*

$$\mu_s(\theta) = \begin{cases} \theta & \text{for } a = 1 \\ \theta^3 + 3\theta & \text{for } a = 3 \end{cases}, \quad \Sigma_s(\theta) = \begin{cases} 1 & \text{for } a = 1 \\ 9\theta^4 + 36\theta^2 + 15 & \text{for } a = 3 \end{cases},$$

which are differentiable w.r.t. θ with derivatives straightforward to calculate. With the Gaussian mechanism, the variance of Y becomes $H_{s,\epsilon,n} = \Sigma_s(\theta)/n + A^{2a}/(n^2\epsilon^2)$.

Figure 1 compares $F(\theta)$ for $a = 1$ and $a = 3$, separately for $\epsilon = 1$ and $\epsilon = \infty$ corresponding to the non-private case, with $n = 100$ and $A = 10$. As it can be observed, while $s(x) = x$ is always better in the non-private case, in the private case (when $\epsilon = 1$) the choice $s(x) = x^3$ seems better for larger values of θ.

Example 2 (Variance of the normal distribution). *Assume that $\mathcal{X} = (-A, A)$ and the assumed population distribution for X is $\mathcal{P}_\theta = \mathcal{N}(0, \theta)$. Consider $s(x) = |x|^a$. So*

$$\mu_s(\theta) = (2\theta)^{a/2} \frac{1}{\sqrt{\pi}} \Gamma\left(\frac{a+1}{2}\right),$$

$$\Sigma_s(\theta) = (2\theta)^a \left[\frac{1}{\sqrt{\pi}} \Gamma\left(\frac{2a+1}{2}\right) - \frac{1}{\pi} \Gamma^2\left(\frac{a+1}{2}\right) \right],$$

which are differentiable w.r.t. θ. With the Gaussian mechanism, we have $H_{s,\epsilon,n} = \Sigma_s(\theta)/n + A^{2a}/(n^2\epsilon^2)$.

Figure 2 compares $F(\theta)$ for various values of a, separately for $\epsilon = 1$ and $\epsilon = \infty$ corresponding to the non-private case, with $n = 100$ and $A = 100$. As it can be observed, while $s(x) = x^2$ is always better in the non-private case, in the private case (when $\epsilon = 1$), the best choice is $s(x) = |x|$.
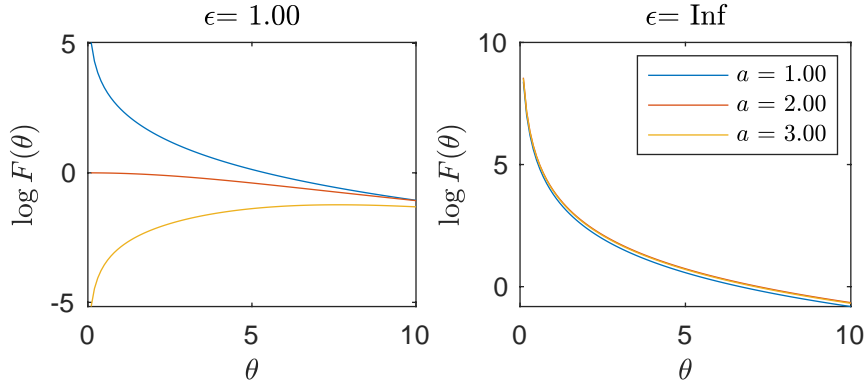
Figure 2: $F(\theta)$ for the variance parameter of $\mathcal{N}(0, \theta)$ when $s(x) = |x|^a$. Left: $\epsilon = 1$, Right: $\epsilon = \infty$ (non-private case).

Example 3 (Width of the uniform distribution). *Let $\mathcal{P}_\theta = \mathrm{Unif}(-\theta, \theta)$ so that 2θ is the width parameter of the uniform distribution. Assume that $s(x) = |x|^a$ for some $a > 0$. We have*

$$\mu_s(\theta) = \frac{\theta^a}{a+1}, \quad \Sigma_s(\theta) = \frac{\theta^{2a} a^2}{(a+1)^2(2a+1)},$$

which are differentiable w.r.t. θ. Assume that $\mathcal{X} = (-A, A)$. Then the sensitivity of s is A^a, hence $\Delta_{s,n} = A^a/n$. This yields that $H_{s,\epsilon,n}(\theta) = \Sigma_s(\theta)/n + A^{2a}/(n^2\epsilon^2)$.

Figure 3 compares $F(\theta)$ for several values of a, separately for $\epsilon = 1$ and $\epsilon = \infty$ corresponding to the non-private case, with $n = 100$ and $A = 100$.
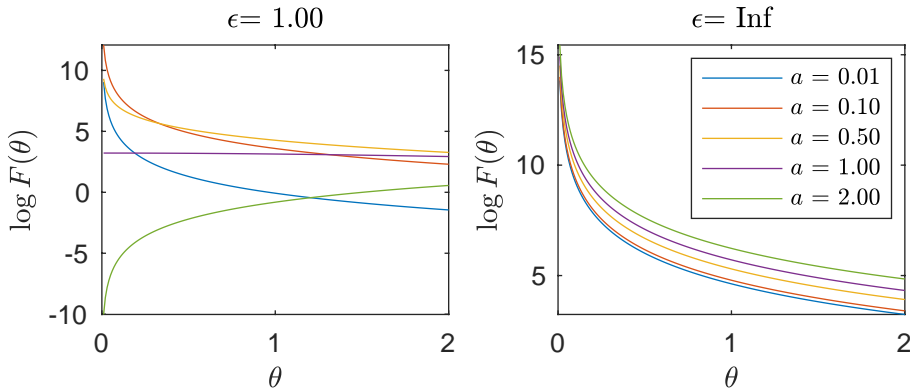

Figure 3: $F(\theta)$ for the width parameter of $\mathrm{Unif}(-\theta, \theta)$ when $s(x) = |x|^a$. Left: $\epsilon = 1$, Right: $\epsilon = \infty$ (non-private case).

Example 3 reveals that while $F(\theta)$ does not exist for the width parameter of the uniform distribution, it does exist for the marginal distribution of Y as long as $\mu_s(\theta) \ \Sigma_s(\theta)$ are differentiable with respec to θ, thanks to the normal approximation of the distribution of the statistic. This is a promising fact for the breadth of models where the proposed methodology for statistic selection applies.

Algorithm 1: Monte Carlo estimation of $F(\theta)$ for (1) - normal approximation for $f_{S_n}(u|\theta)$.

Input: θ: parameter; n: data size; N, M: Monte Carlo parameters

Output: $\widehat{F(\theta)}$: Estimate of $F(\theta)$

1 for $i = 1, \ldots, N$ **do**

2   Sample $y^{(i)} \sim p_{\epsilon,S_n}(y|\theta)$

3   **for** $j = 1, \ldots, M$ **do**

4   Sample $u^{(j)} \sim q_\theta(\cdot)$, calculate

$$w_j = \frac{f_{S_n}(u^{(j)}|\theta)g_{\epsilon,S_n}(y^{(i)}|u^{(j)})}{q_\theta(u^{(j)})}$$

         using (11).

5   Using (11), calculate

$$\widehat{\gamma}_{\epsilon,S_n}(\theta; y^{(i)}) = \sum_{j=1}^{N} \frac{\partial \log f_{S_n}(u^{(j)}|\theta)}{\partial \theta} \frac{w_j}{\sum_{j'=1}^{N} w_{j'}}.$$

6 return $\widehat{F(\theta)} = \frac{1}{N} \sum_{i=1}^{N} \widehat{\gamma}_{\epsilon,S_n}(\theta; y^{(i)})\widehat{\gamma}_{\epsilon,S_n}(\theta; y^{(i)})^T$.

3.2   Fisher information with additive statistics and non-gaussian noise

In the previous section, the Gaussian mechanism enabled us to perform an analytical comparisons between statistics. For other privacy preserving mechanisms, comparisons based on $F(\theta)$ can still be made in the same spirit, however by using Monte Carlo estimates of $F(\theta)$, as we will see next.

A typical example to a non-gaussian mechanism is the Laplace mechanism. In the Laplace mechanism to provide ϵ-DP, privacy preserving noise in (1) is distributed according to

$$V \sim \text{Laplace}(\Delta_{s,1}/(n\epsilon)).$$

As long as S_n is an additive statistic, we can employ the normal approximation in (8) for its distribution. Even so, the approximation of $F(\theta)$ given in Section 3.1 may not be accurate when non-gaussian noise is used to preserve privacy. Furthermore, the integral in (4) will be typically intractable, as well as the derivative of its logarithm. As a result, it may also be difficult to calculate $F(\theta)$ exactly. Fortunately, a consistent Monte Carlo estimator of $F(\theta)$ is available. We present its details in the following.

Define the variable $U = S_n(X_{1:n})$, and let $f_{S_n}(u|\theta)$ be the probability density of U given θ evaluated at u. In most of the models considered in this paper, the conditional distribution $p_{\epsilon,S_n}(y|x_{1:n})$ depends only on $u = S_n(x_{1:n})$. (See the discussion about smooth sensitivity in Section 3.3 for an exception to this.). If that is the case, we can define $g_{\epsilon,S_n}(y|u)$ to be the density of the conditional distribution of Y given U calculated at $Y = y, U = u$. Then the

marginal distribution can also be written as

$$p_{\epsilon,S_n}(y|\theta) = \int g_{\epsilon,S_n}(y|u)f_{S_n}(u|\theta)du. \qquad (10)$$

Based on (10), the Fisher's identity for the score vector can be written as

$$\gamma_{\epsilon,S_n}(\theta;y) = \int \frac{\partial \log f_{S_n}(u|\theta)}{\partial \theta} p(u|y,\theta)du.$$

where the integral is taken with respect to the posterior distribution

$$p(u|y,\theta) \propto f_{S_n}(u|\theta)g_{\epsilon,S_n}(y|u).$$

The Monte Carlo estimation of $F(\theta)$ is based on estimating the above integral via importance sampling, exact sampling (e.g. via rejection sampling) or approximate sampling (via MCMC) from $p(u|y,\theta)$. Once we have a method for obtaining a numerical approximation of the score vector at given y and θ, $F(\theta)$ can be estimated according to (6).

A Monte Carlo estimator of $F(\theta)$ in the presence of additive statistic and non-gaussian noise is given in Algorithm 1. The estimator is based on the estimation of the score vector using self-normalised importance sampling with proposal distribution $q_\theta(u)$. Further, the normal approximation in (8) is employed, enabling

$$f_{S_n}(u|\theta) = \mathcal{N}(u;\mu_s(\theta),\Sigma_s(\theta)/n) \qquad (11)$$

in the calculations. Sampling from $p_{\epsilon,S_n}(y|\theta)$ can be performed straightforwardly since the model for Y is generative. Also, the importance sampling stage (the inner loop) can be replaced by a MCMC routine to collect M samples with equal weights for u from the conditional distribution $p(u|y,\theta)$ and estimate the score by $\frac{1}{M}\sum_{j=1}^{M} \frac{\partial \log f_{S_n}(u|\theta)}{\partial \theta}$.

3.3   Fisher information based on the true marginal distribution

Note that Algorithm 1 exploits the normal approximation in (8) for the statistic $S(X_{1:n})$. In some cases, this approximation may be unavailable or unjustifiable. This may be because $S_n(X_{1:n})$ is a *non-additive* statistic, or the moments $\mu(\theta)$ and $\Sigma(\theta)$ are intractable.

In such cases, one can still devise a Monte Carlo method to estimate $F(\theta)$ based on the true marginal distribution of the observed (noisy) statistic in (1). Such a method is given in Algorithm 2. Algorithm 2 exploits (4), which expresses the marginal distribution in terms of $X_{1:n}$. The reason we resorted to $X_{1:n}$ instead of $U = S_n(X_{1:n})$ is that in this part we are concerned with a setting where the probability distribution of U is hard to find or approximate. Accordingly, the algorithm samples $X_{1:n}$ from their population distribution and uses importance sampling to calculate the score vector as an expectation of the derivative of the log-joint density of $(X_{1:n},Y)$ with respect to the posterior distribution of $X_{1:n}$ given Y. As a result a requirement is that the population distribution is differentiable with respect to θ.

At this point it is worth pointing to smooth sensitivity (Nissim et al., 2007), a method that has proven quite useful in reducing privacy preserving noise considerably, especially

---

Algorithm 2: Monte Carlo estimation of Fisher information for (1) - exact marginal distribution

---

Input: θ: parameter; n: data size; N, M: Monte Carlo parameters

Output: $\widehat{F(\theta)}$: Estimate of $F(\theta)$

1 for $i = 1, \ldots, N$ **do**

2   Sample $y^{(i)} \sim p_{\epsilon, S_n}(y|\theta)$

3   **for** $j = 1, \ldots, M$ **do**

4   **for** $t = 1, \ldots, n$ **do**

5   Sample $x_t^{(j)} \sim p(x|\theta)$.

6   Set $w_j = p_{\epsilon, S_n}(y^{(i)}|x_{1:n}^{(j)})$.

7   Calculate

$$\widehat{\gamma}_{\epsilon, S_n}(\theta; y^{(i)}) = \sum_{j=1}^{N} \left(\sum_{t=1}^{n} \frac{\partial \log p(x_t^{(j)}|\theta)}{\partial \theta} \right) \frac{w_j}{\sum_{j'=1}^{N} w_{j'}}.$$

8 return $\widehat{F(\theta)} = \frac{1}{N} \sum_{i=1}^{N} \widehat{\gamma}_{\epsilon, S_n}(\theta; y^{(i)}) \widehat{\gamma}_{\epsilon, S_n}(\theta; y^{(i)})^T.$

---

for non-additive statistics, which are under consideration in this section. As before, one could use the global sensitivity of S_n as in Definition 2 to determine the amount of noise to generate Y. However, for some non-additive statistics, such as max and median, adding noise based on the global sensitivity can be quite ineffective. This is because the global sensitivity of the those functions is as large as the range of S_n. For example, if $\mathcal{X} \mapsto [0, A]$, the global sensitivites of max and median are both A. Instead, one can generate the noisy statistic Y by adjusting the amount of noise using the smooth sensitivity defined in Nissim et al. (2007).

Definition 3 (Smooth sensitivity). *For a function $\psi : \mathcal{X}^n \mapsto \mathbb{R}^{d_\psi}$ and $\beta > 0$, define the β-smooth sensitivity as*

$$\Delta_{\psi, \beta}^{smooth}(x_{1:n}) = \max_{x_{1:n}' \in \mathcal{X}^n} L_\psi(x_{1:n}') e^{-\beta h(x_{1:n}, x_{1:n}')},$$

where $L_\psi(x_{1:n})$ is called the local sensitivity at $x_{1:n}$ and defined as

$$L_\psi(x_{1:n}) = \max_{x_{1:n}' : h(x_{1:n}, x_{1:n}') = 1} \|\psi(x_{1:n}) - \psi(x_{1:n}')\|_1.$$

Differential privacy can be provided based on local sensitivity using appropriate noise-adding mechanisms. For example, to satisfy (ϵ, δ)-DP for $\epsilon, \delta \in (0, 1)$, one can generate $Y = S_n(X_{1:n}) + V$ with

$$V \sim \text{Laplace}\left(\Delta_{S_n, \beta}^{smooth}(X_{1:n})/\alpha\right),$$

where $\alpha = \epsilon/2$ and $\beta = \epsilon/[2\ln(2/\delta)]$. (Pure differential privacy, i.e., with $\delta = 0$, can also be obtained using smooth sensitivity, however with a noise distribution whose tails decay slower than exponentially, such as Cauchy distribution.)

13

Algorithm 3: Monte Carlo estimation of $F(\theta)$ for (3)

Input: θ: parameter; n: data size; N, M: Monte Carlo parameters

Output: $\widehat{F(\theta)}$: Estimate of $F(\theta)$

1 for $i = 1, \ldots, N$ **do**

2 $\quad$ Sample $y^{(i)} \sim p_{\epsilon,s}(y|\theta)$

3 $\quad$ **for** $j = 1, \ldots, M$ **do**

4 $\quad\quad$ Sample $x^{(j)} \sim q_\theta(x)$ and calculate

$$w_j = p(x^{(j)}|\theta) g_{\epsilon,s}(y^{(i)}|s(x^{(j)}))/q_\theta(x^{(j)}).$$

5 $\quad$ Calculate

$$\widehat{\gamma}_{\epsilon,s}(\theta; y^{(i)}) = \sum_{j=1}^{N} \frac{\partial \log p(x^{(j)}|\theta)}{\partial \theta} \frac{w_j}{\sum_{j'=1}^{N} w_{j'}}.$$

6 return $\widehat{F(\theta)} = \frac{n}{N} \sum_{i=1}^{N} \widehat{\gamma}_{\epsilon,S_n}(\theta; y^{(i)}) \widehat{\gamma}_{\epsilon,S_n}(\theta; y^{(i)})^T$.

Note that, contrary to the earlier examples, using smooth sensitivity determines the noise distribution dependent on $X_{1:n}$, rendering a quite non-standard joint distribution, in particular a non-standard posterior distribution for the parameter of interest θ. This highlights the importance of general-purpose inference methods in the privacy context such as MCMC.

Finally, a remark on the notation. When smooth sensitivity is used, the density of the conditional distribution y given $X_{1:n} = x_{1:n}$ depends on not only $S_n(x_{1:n})$ but also $x_{1:n}$ itself, since $x_{1:n}$ determines the noise variance also. To cover those cases, in Algorithm 2 we resort the more general representation $p_{\epsilon,S_n}(y^{(i)}|x_{1:n}^{(j)})$ to denote the conditional distribution of y given $x_{1:n}$.

3.4 Fisher information with sequential release

In Sections 3.1-3.3 we looked at scenarios where a single statistic of the sensitive data $X_{1:n}$ is shared. Alternatively, private data can be sequentially released as $Y_1, \ldots, Y_n$, where each Y_i is a noisy version of $s(X_i)$ as in (3). This corresponds to a scenario where the analyst collects data from the individuals separately in a privacy preserving way. The former and the latter models are also referred to as the centralized model and the local model (Kasiviswanathan et al., 2008), respectively. The local model comes with the expense of adding much more noise to each Y_i than the statistic $S(X_{1:n})$. Specifically, to provide ϵ-DP with the Laplace mechanism, we must have

$$Y_i = s(X_i) + V_i, \quad V_i \sim \text{Laplace}(\Delta_{s,1}/\epsilon)$$

which no longer has the $1/n$ factor in its noise parameter.

In the local model, we can talk about the marginal distribution of each Y_i, whose

probability density can be written as

$$p_{\epsilon,s}(y|\theta) = \int p(x|\theta)g_{\epsilon,s}(y|s(x))dx, \tag{12}$$

where $g_{\epsilon,s}(y|s(x))$ is the probability density function of the conditional distribution of Y given $S(X)$, which, according to (3), reduces to the probability of $\mathcal{P}_{\epsilon,s}$ evaluated at $y - s(x)$.

The Fisher information corresponding to this mechanism can be numerically calculated by estimating the Fisher infrormation of a single Y_i via Monte Carlo as in Algorithm 3. The algorithm requires that the probability density (mass) function of X_i is differentiable w.r.t. θ.

Example 4 (Binary responses). *Let $X_i \in \{0, 1\}$ with $X_i \overset{iid}{\sim} Bern(\theta)$ for $i = 1, \ldots, n$. In a non-private setting, a natural estimator for θ is $\bar{X}$. Instead, we consider estimating θ privately. We will compare three mechanisms.*

1. *It is well known, and can be easily verified that releasing the randomized binary responses $Y_1, \ldots, Y_n$, where*

$$Y_i = \begin{cases} X & \text{with probability}\frac{e^\epsilon}{e^\epsilon+1} \\ 1 - X_i & \text{else} \end{cases}$$

   provides ϵ-DP. The probability of the randomized response being 1 is given by

$$\tau := \mathbb{P}(Y = 1) = \frac{\theta e^\epsilon + (1 - \theta)}{1 + e^\epsilon}.$$

   The probability density of Y given θ and ϵ is

$$\log p(y|\theta) = y \ln \tau + (1 - y) \ln \tau.$$

   Therefore, letting $\alpha = (e^\epsilon - 1)/(e^\epsilon + 1)$, the Fisher information of $Y_1, \ldots, Y_n$ is given by

$$F_1(\theta) = n\mathbb{E}\left[-\frac{\partial^2 \log p(Y|\theta)}{\partial \theta^2}\right] = \frac{n\alpha^2}{\tau(1 - \tau)}.$$

2. *One alternative to the above is to release $Z_i = X_i + V_i$, with $V_i \overset{i.i.d}{\sim} \mathcal{N}(0, 1/\epsilon^2)$. It is obvious that $Z_1, \ldots, Z_n$ is as informative as $\hat{\theta}_2 = \bar{Z}$, which approximately has the normal distribution $\mathcal{N}(\theta, \theta(1 - \theta)/n + 1/(\epsilon^2 n))$ hence its Fisher information is approximately*

$$F_2(\theta) = \frac{n(\theta(1 - \theta) + 1/\epsilon^2) + (1 - 2\theta)^2}{[\theta(1 - \theta) + 1/\epsilon^2]^2}.$$

3. *Finally, we consider adding noise to the mean $\bar{X}$ and obtain $\hat{\theta}_3 = \bar{X} + V$, where $V \sim \mathcal{N}(0, 1/(n^2\epsilon^2))$. Therefore, $\hat{\theta}_3 \sim \mathcal{N}(\theta, \theta(1-\theta)/n + 1/(\epsilon^2 n^2))$. This last estimator is based on a noisy average whose Fisher information is*

$$F_3(\theta) = \frac{n(\theta(1 - \theta) + 1/(\epsilon^2 n)) + (1 - 2\theta)^2}{[\theta(1 - \theta) + 1/(\epsilon^2 n)]^2}.$$

   Notice the improvement due to adding noise to the average (output) rather than averaging noisy inputs.

Figure 4 shows a comparison between $F_1(\theta)$ and $F_2(\theta)$ as well as between $F_1(\theta)$ and $F_3(\theta)$ for $n = 100$. It can be seen that, for small values of ϵ, revealing the average of the randomizing the responses is better than revealing the average of the noisy responses created by the Gaussian mechanism. However, in the same ϵ regimes, using the noisy average is better than the average of the randomised responses.
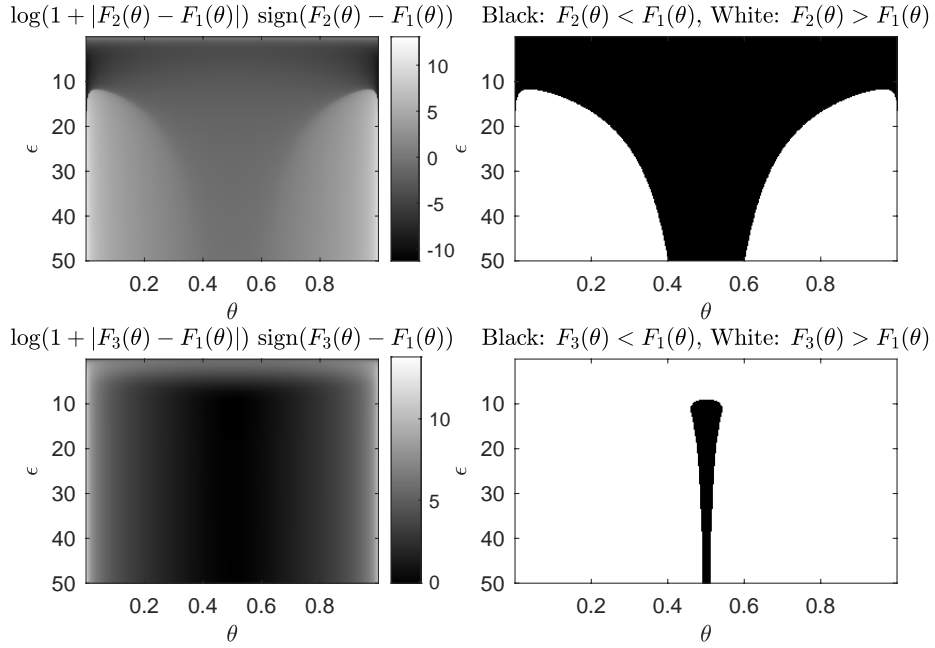

Figure 4: Comparison among $F_1(\theta)$, $F_2(\theta)$, $F_3(\theta)$.

Graphical summary: Figure 5 shows graphical representations of the models respected by the $F(\theta)$ calculations in this section. Note that the graphs (1-3) correspond to the same batch model in (1), but represented with different sets of variables, while graph (4) corresponds to the model with sequential release in (3). Moreover, the graphs (1-4) correspond to the MCMC algorithms in Sections 4.1-4.4, respectively.
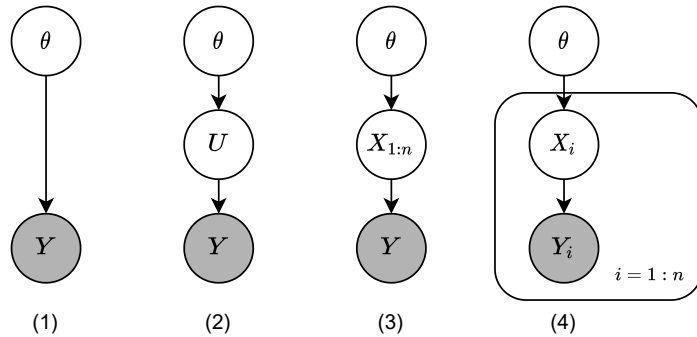

Figure 5: Graphical representations of the models respected in the $F(\theta)$ calculations and MCMC algorithms. Models (1-4) refer to Sections 3.1-3.4, respectively, and correspond to the MCMC algorithms in Sections 4.1-4.4, respectively. Shaded variables are observed; the others are latent.

Table 2: Algorithm-model matching for differentially private Bayesian learning via MCMC

Model	Algorithm
Additive statistic, Gaussian noise	Algorithm 4
Additive statistic, non-gaussian noise	Algorithm 5, 6
Non-additive statistic	Algorithm 7
Sequential release	Algorithm 8

4 Bayesian inference using MCMC

For the statistic selection method to be useful in practice, it should be accompanied with an inference method. Within the Bayesian framework, one could use ideas from approximate Bayesian computation. This relation is already observed in Gong (2019), where an EM algorithm is presented for maximum likelihood estimation of θ. The EM algorithm can be somewhat restrictive, for its E- and M- steps may require exact posterior expectations. An alternative to EM is to consider Bayesian estimation by means of sampling from the posterior distribution of θ given the shared statistics. Owing to the availability of Monte Carlo techniques for sampling from various forms of posterior distributions, Bayesian estimation is usually less demanding about the nature of the model in hand.

In the batch setting, where a statistic $S_n(X_{1:n})$ is shared in noise as in (1), the posterior distribution is

$$p_{\epsilon,S_n}(\theta|y) \propto \eta(\theta)p_{\epsilon,S_n}(y|\theta), \tag{13}$$

where $\eta(\theta)$ is the probability density of the prior distribution of θ and the likelihood $p_{\epsilon,S_n}(y|\theta)$ is defined in (10). In the sequential setting, where a function s of each X_i is shared in noise, the posterior distribution becomes

$$p_{\epsilon,s}(\theta|y_{1:n}) \propto \eta(\theta)\prod_{t=1}^{n} p_{\epsilon,s}(y_t|\theta). \tag{14}$$

In the following, we propose MCMC algorithms for sampling from the posterior distribution for the settings investigated separately in the subsections of Section 3.

MCMC is the name for a family of methods that (approximately) sample from a given probability distribution, say $\pi(\theta)$. An MCMC algorithm is specified by an ergodic Markov chain $\{\theta_i\}_{i\geq 0}$ which is designed to have $\pi(\theta)$ as its invariant distribution. In that way, the generated sequence $\{\theta_i\}_{i\geq 0}$ from this Markov chain converges in distribution to $\pi(\theta)$. The methods we propose in this paper are either variants or sophisticated imitations of the Metropolis-Hastings (MH), arguably the most popular MCMC algorithm. One iteration of the MH algorithm involves (i) a proposal mechanism where, given the current value $\theta_i = \theta$, a candidate value θ' is proposed from the conditional distribution $q(\theta'|\theta)$, and (ii) an accept-reject mechanism in which the proposal is accepted and $\theta_{i+1} = \theta'$ is taken with the acceptance probability

$$\alpha(\theta, \theta') = \min\left\{1, \frac{q(\theta|\theta')}{q(\theta'|\theta)}\frac{\pi(\theta')}{\pi(\theta)}\right\};$$

and otherwise it is rejected and the new sample is taken as the current sample, i.e., $\theta_{i+1} = \theta$.

In the following subsections, we will propose MH-based algorithms that are suitable for each data-sharing model investigated in Sections 3.1-3.4.

4.1 MH for additive statistic and Gaussian noise

Recall that when $S(X_{1:n})$ is an additive statistic as in (2) and the noise is Gussian, the marginal likelihood of Y given θ can be approximated as (9). We can use that approximation to obtain

$$\hat{p}_{\epsilon,S_n}(\theta|y) \propto \eta(\theta)\mathcal{N}(y; \mu_s(\theta), H_{s,\epsilon,n}(\theta)). \tag{15}$$

If this distribution is intractable, MCMC can be used to sample from it. Algorithm 4 presents the MH algorithm for this distribution.

Algorithm 4: MH for (15) - one iteration

Input: Current value: θ; privately shared statistic: y, privacy level: ϵ
Output: New sample
1 Propose $\theta' \sim q(\cdot|\theta)$
2 Accept the proposal and return θ' with probability

$$\min\left\{1, \frac{q(\theta|\theta')}{q(\theta'|\theta)}\frac{\eta(\theta')}{\eta(\theta)}\frac{\mathcal{N}(y; \mu_s(\theta'), H_{s,\epsilon,n}(\theta'))}{\mathcal{N}(y; \mu_s(\theta), H_{s,\epsilon,n}(\theta))}\right\};$$

otherwise reject the proposal and return θ.

4.2 MH for additive statistic and non-gaussian noise

Here we study inference for the setting discussed in Section 3.2, where $S_n(X_{1:n})$ is still additive as in (2), however a non-Gaussian mechanism (such as the Laplace mechanism) is used to preserve privacy.

Due to the additivity of S_n, we can still use the normality approximation in (8) for $U = S_n(X_{1:n})$. However, due to non-gaussianity of the noise, the marginal distribution of the shared statistic Y may not reliably be approximated as a normal distribution any more.

In this model, inference can still be made via suitable MCMC algorithms. Define the joint posterior distribution

$$\pi(\theta, u|y) \propto \eta(\theta)f_{S_n}(u|\theta)g_{\epsilon,S_n}(y|u) \tag{16}$$

where, recall that, $g_{\epsilon,n}(y|u)$ is the conditional distribution of y given $u = S_n(x_{1:n})$. We consider sampling from this posterior distribution by using MCMC. Note that the marginal distribution of θ with respect to $\pi(\theta, u|y)$ can be shown to be $p_{\epsilon,S_n}(\theta|y)$ in (13), which validates sampling from $\pi(\theta, u|y)$ as a means of sampling from $p_{\epsilon,S_n}(\theta|y)$.

There are many possible ways to design a correct MCMC algorithm for $\pi(\theta, u|y)$. A standard option is to use the MH-within-Gibbs algorithm, where one iteration consists of

an update of u conditional on θ, y which is followed by an update of θ conditional on u, y. The MH-within-Gibbs algorithm may not be efficient in the presence of high dependence between the variables θ and u given y.

Alternative to MH-within-Gibbs, exact-approximate MCMC algorithms (Andrieu and Roberts, 2009; Andrieu et al., 2010, 2020) mimic the MH algorithm for the marginal posterior distribution in (13). The term "exact-approximate" comes from the fact that the Markov chains of those algorithms still correctly converge to the exact posterior distribution (hence "exact") and they are approximations of the ideal (but intractable) MH algorithm for the marginal posterior distribution $p_{\epsilon, S_n}(\theta|y)$ (hence "approximate"). Those algorithms can be useful since they circumvent the problem of dependency between θ and u by relying on sample-based estimators of the marginal MH acceptance ratio. The variance of the estimator reduces with amount of computation. Moreover, the amount of computation can be mostly parallelized.

In the following we present two exact-approximate MCMC algorithms.

4.2.1 Pseudo-Marginal MH

The pseudo-marginal MH (PMMH) of Andrieu and Roberts (2009), adopted to the posterior distribution in (13) is described in Algorithm 5. The PMMH algorithm targets the posterior distribution in (16), but it mimics the MH algorithm by estimating the intractable marginal likelihood (10) in (13) using importance sampling. Observe that the computational cost of one iteration of this algorithm is $\mathcal{O}(N)$, the sample size of the importance sampling step, which can largely be parallelised.

Algorithm 5: PMMH for the posterior distribution in (16) - one iteration

Input: Current sample: $(\theta, \hat{Z})$, number of proposals for u: N privately shared statistic y

Output: New sample

1 Propose $\theta' \sim q(\cdot|\theta)$
2 Sample $u^{(j)} \sim q_{\theta'}(\cdot)$ for $j = 1, \ldots, N$.
3 Calculate $\hat{Z}' = \frac{1}{N} \sum_{j=1}^{N} f_{S_n}(u^{(j)}|\theta) g_{\epsilon,n}(y|u^{(j)})/q_{\theta'}(u^{(j)})$ using (11).
4 Return $(\theta', \hat{Z}')$ with probability

$$\min\left\{1, \frac{q(\theta|\theta')}{q(\theta'|\theta)} \frac{\eta(\theta')}{\eta(\theta)} \frac{\hat{Z}'}{\hat{Z}}\right\};$$

otherwise, reject and return $(\theta, \hat{Z})$.

4.2.2 MH with Averaged Acceptance Ratios

In PMMH, the estimate $\hat{Z}$ in the denominator of the acceptance ratio is carried over from the previous iteration, which can lead to stickiness in its Markov chain. The correlated pseudo-marginal algorithm of Deligiannidis et al. (2018) partially alleviates the stickiness problem

by making the numerator and denominator correlated, which is achieved by employing a common source of randomness in the estimators of the the numerator and denominator of the marginal acceptance ratio. This idea of using correlated estimators is taken to its limit by a more recent class of exact-approximate MCMC algorithms proposed in Andrieu et al. (2020) with the name "MH with Averaged Acceptance Ratio (MHAAR)". Unlike PMMH or its correlated version, in MHAAR both the numerator and the denominator of the acceptance ratio estimator are (almost) fully refreshed in every iteration, which is one advantage of MHAAR over PMMH.

While there are several versions of MHAAR which can be applied to the posterior distribution in (16), we present a particular variant in Andrieu et al. (2020, Section 3) in Algorithm 6. The requirement in Algorithm 6 to work properly, the proposal distribution for u has to satisfy $q_{\theta,\theta'}(u) = q_{\theta',\theta}(u)$ for all θ, θ' and u.

Algorithm 6: MHAAR for the posterior distribution in (16) - one iteration

Input: Current value: (θ, u); number of proposals for u: N; privately shared statistic: y

Output: New sample

1 Propose $\theta' \sim q(\cdot|\theta)$

2 **for** $j = 1, \ldots, N$ **do**

3 $\quad$ If $j = 1$ set $u^{(1)} = u$; otherwise sample $u^{(j)} \sim q_{\theta,\theta'}(\cdot)$.

4 $\quad$ Using (11), calculate

$$w_j = \frac{f_{S_n}(u^{(j)}|\theta) g_{\epsilon,n}(y|u^{(j)})}{q_{\theta,\theta'}(u^{(j)})}, \quad w_j' = \frac{f_{S_n}(u^{(j)}|\theta') g_{\epsilon,n}(y|u^{(j)})}{q_{\theta,\theta'}(u^{(j)})}$$

5 With probability

$$\min\left\{ 1, \frac{q(\theta|\theta')}{q(\theta'|\theta)} \frac{\eta(\theta')}{\eta(\theta)} \frac{\sum_{j=1}^{N} w_j'}{\sum_{j=1}^{N} w_j} \right\},$$

sample $k \in \{1, \ldots, N\}$ with probability proportional to w_k' and return $(\theta', u^{(k)})$. Otherwise, reject the move, sample $k \in \{1, \ldots, N\}$ with probability proportional to w_k, and return $(\theta, u^{(k)})$.

4.3 Exact inference based on the true posterior

The algorithms in the previous sections can be restrictive, since $\mu_s(\theta)$ and $\Sigma_s(\theta)$ may not be analytically available, or the normality approximation may not be valid if S_n is an extreme statistic of $X_{1:n}$, such as $S(x_{1:n}) = \max_{1 \leq t \leq n} s(x_t)$. In such models, the true posterior of θ may be targeted via a special variable augmentation. Consider, for example, the extended posterior distribution

$$\pi(\theta, x_{1:n}|y) \propto \eta(\theta) p(x_{1:n}|\theta) p_{\epsilon,S_n}(y|x_{1:n}).$$

(Alternatively, one may choose to augment the space with the statistic $u = S(x_{1:n})$ and work with (16) if $f_{S_n}(u|\theta)$ can be calculated exactly. However we do not pursue this option to avoid diverting from the main point.)

One can go to an even lower-level representation and express the joint distribution in terms of the random variables that generate X_i's and have distributions that do not depend on θ. Letting those random variables $Z_i \sim \mu(\cdot)$, assume a transformation $\varphi_\theta(\cdot)$ such that if

$$Z_i \overset{\text{i.i.d}}{\sim} \mu(\cdot) \Rightarrow X_i = \varphi_\theta(Z_i) \overset{\text{i.i.d}}{\sim} \mathcal{P}_\theta, \quad i \geq 1 \tag{17}$$

Without loss of generality, $Z_{1:n}$ can be thought of a sequence of random variables from $\text{Unif}(0,1)$, owing to the role of uniformly distributed pseudo-random variables in generation of random variables from any distribution via some suitable transformation.

Presence of such $Z_{1:n}$ induces the joint posterior distribution

$$\pi(\theta, z_{1:n}|y) \propto \eta(\theta) \prod_{t=1}^{n} \mu(z_t) h_{\epsilon, S_n}(y|z_{1:n}, \theta), \tag{18}$$

where $h_{\epsilon, S_n}(y|z_{1:n}, \theta) = p_{\epsilon, S_n}(y|x_{1:n})$, with $x_i = \varphi(x_i)$, is a re-parametrization of the conditional density in terms of $z_{1:n}$. Crucially, it can be shown that the marginal distribution for θ with respect to $\pi(\theta, z_{1:n}|y)$ is the target posterior distribution $p_{\epsilon, S_n}(\theta|y)$.

Choosing $Z_{1:n}$ such that its density does not depend on θ enables the MHAAR methodology of Andrieu et al. (2020), where estimates of the acceptance ratio can efficiently be averaged to reduce variance. Algorithm 7 is inspired from Andrieu et al. (2020) and can be thought of a variant of MHAAR. However, it bears methodological novelty in the sense that, if desired, only a subset of the latent variables $z_{1:n}$ can be updated per iteration instead of the whole $z_{1:n}$ (which may be computationally cheap in some cases.) Hence, Algorithm 7 requires to be proven for its validity. We establish that in the following proposition. A proof is given in Appendix A.

Proposition 1. *The Markov kernel of Algorithm 7 has $\pi(\theta, z_{1:n}|y)$ in (18) as its invariant distribution.*

4.4 Exact inference based on sequential releases

Here we consider the scenario in Section 3.4, where the individuals' data are obtained in privacy preserving noise as in (3). Here we have independent sequential observations Y_i, whose generative model involves a latent variable, X_i. As in Section 4.3, we will adopt the representation of the generative model in terms of random variables Z_i whose distribution does not depend on θ. Specifically, we assume that (17) holds for some distribution $\mu(\cdot)$ and functions $\varphi_\theta(\cdot)$ for all θ. Again, as long as one can sample from $\mathcal{P}_\theta$, existence of such Z is secured. This enables the joint distribution

$$\pi(\theta, z_{1:n}|y_{1:n}) \propto \eta(\theta) \prod_{t=1}^{n} \mu(z_t) h_\epsilon(y_t|z_t, \theta), \tag{19}$$

Algorithm 7: MHAAR for (18) - one iteration

Input: Current sample: $(\theta, z_{1:n})$, subset size: $m < n$, number of samples for $z_{1:n}$: N, privately shared statistic: y.

Output: New sample

1 Propose $\theta' \sim q(\cdot|\theta)$

2 if *full mode* **then**

3 $\quad$ Set $z_{1:n}^{(1)} = z_{1:n}$ and propose $z_{1:n}^{(2)}, \ldots, z_{1:n}^{(N)} \sim \mu(\cdot)$.

4 else

5 $\quad$ Set $z_{1:n}^{(1)} = z_{1:n}$.

6 $\quad$ Sample (without replacement) a subset $b = \{b_1, \ldots, b_m\} \subset \{1, \ldots, n\}$ uniformly.

7 $\quad$ **for** $i = 2, \ldots, N$ **do**

8 $\quad\quad$ Set $z_{/b}^{(i)} = z_{/b}$, propose $z_b^{(i)} \sim \prod_{i=1}^m \mu_{b_i}(\cdot)$, and set $z_{1:n}^{(i)} = (z_b^{(i)}, z_{/b})$.

9 Sample k with probability proportional to $h_\epsilon(y|z_{1:n}^{(k)}, \theta')$.

10 Accept $\theta', z_{1:n}^{(k)}$ as the new sample with probability

$$\min\left\{1, \frac{q(\theta|\theta')}{q(\theta'|\theta)} \frac{\eta(\theta')}{\eta(\theta)} \frac{\sum_{i=1}^N h_\epsilon(y|z_{1:n}^{(i)}, \theta')}{\sum_{i=1}^N h_\epsilon(y|z_{1:n}^{(i)}, \theta)}\right\};$$

otherwise reject and repeat $(\theta, z_{1:n})$ as the new value.

---

where $h_\epsilon(y_t|z_t, \theta) = g_\epsilon(y_t|s(\varphi_\theta(z_t)))$. We aim to sample from the posterior distribution in (19) using Algorithm 8, which we adopt from a recently developed MHAAR algorithm in Andrieu et al. (2020). The reason we choose this particular algorithm is that the variance of its acceptance ratio does not increase with n as long as the proposal distribution for θ is properly scaled with the data size n; see Yıldırım et al. (2018) for a related result. Note that this is in contrast to the PMMH algorithm of Andrieu and Roberts (2009) whose acceptance rate increases with n, leading to higher rejection rates, hence to slowing of the algorithm.

5   Numerical examples

In this section we show some numerical examples which justify the proposed way of choosing the statistic of the sensitive data, as well as demonstrate the performance of the MCMC algorithms proposed for the different privacy settings that are described in Section 3.

For Bayesian inference, a method for statistic selection can be reasonably justified if it yields the statistic that results in smallest MSE for the posterior expectation $\hat{\theta}(Y) = \mathbb{E}(\theta|Y)$, that is,

$$\text{MSE} = \mathbb{E}_Y[(\hat{\theta}(Y) - \theta^*)^2], \tag{20}$$

In our experiments we will follow that way of justification for our statistic selection method based on the Fisher information. For a given y, $\hat{\theta}(y)$ will be obtained by one of the MCMC algorithms presented in Section 4, depending on the nature of the data generation model.

Algorithm 8: MHAAR for (19) - one iteration

Input: Current sample $(\theta, z_{1:n})$, subset size $m < n$, number of samples for $z_{1:n}$: N, privately shared sequence: $y_{1:n}$.

Output: New sample

1 Propose $\theta' \sim q(\cdot|\theta)$

2 for $t = 1, \ldots, n$ do

3 $\quad$ Set $z_t^{(1)} = z_t$ and propose $z_t^{(2)}, \ldots, z_t^{(N)} \sim \mu(\cdot)$.

4 Calculate the acceptance probability

$$\alpha = \min\left\{1, \frac{q(\theta|\theta')}{q(\theta'|\theta)} \frac{\eta(\theta')}{\eta(\theta)} \frac{\prod_{t=1}^n \sum_{i=1}^N h_\epsilon(y_t|z_t^{(i)}, \theta')}{\prod_{t=1}^n \sum_{i=1}^N h_\epsilon(y_t|z_t^{(i)}, \theta)}\right\}.$$

5 Sample $v \sim \text{Unif}(0, 1)$.

6 if $v < \alpha$ then

7 $\quad$ Return $(\theta', z_{1:n} = (z_1^{(k_1)}, \ldots, z_n^{(k_n)}))$, where each $k_t \in \{1, \ldots, N\}$ is sampled with probability proportional to $h_\epsilon(y_t|z_t^{(k_t)}, \theta')$.

8 else

9 $\quad$ Return $(\theta, z_{1:n} = (z_1^{(k_1)}, \ldots, z_n^{(k_n)}))$, where each $k_t \in \{1, \ldots, N\}$ is sampled with probability proportional to $h_\epsilon(y_t|z_t^{(k_t)}, \theta)$.

MSE in (20) will approximated by MSE $\approx \frac{1}{M} \sum_{i=1}^M (\hat{\theta}(Y^{(i)}) - \theta^*)^2$, where the M independent samples for $Y^{(i)}$ are drawn conditional on the true value θ^*.

5.1 Comparison of additive statistics with the Gaussian mechanism

Our first example refers to the setting in Example 2, where $\mathcal{P}_\theta = \mathcal{N}(0, \theta)$ with the natural limits $[-A, A]$ for the data and we computed $F(\theta)$ when the noisy statistic in (1) is constructed from $s(x) = |x|$ and $s(x) = x^2$, separately. In Example 2 we showed that $s(x) = |x|$ results in larger $F(\theta)$ than $s(x) = x^2$ when $\epsilon = 1$, while $s(x) = x^2$ becomes more informative when there is no privacy.

Here we compared the choices $s(x) = |x|$ and $s(x) = |x^2|$ in terms of MSE at various values of ϵ. We took $A = 10$, $n = 100$, and $\theta^* = 2$. For MSE calculations, we took $M = 10^3$. To obtain the posterior expectations, we ran Algorithm 4, with flat prior on θ, to generate a total of $K = 10^5$ iterations and took the sample average after discarding the first quarter as burn-in.

The results are summarized in Figure 6. We observe that $s(x) = |x|$ outperforms $s(x) = x^2$ in terms of MSE unless ϵ is very large. Critically, we observe that when $F(\theta)$ is larger we have smaller MSE, which justifies the use of Fisher information for statistic selection.
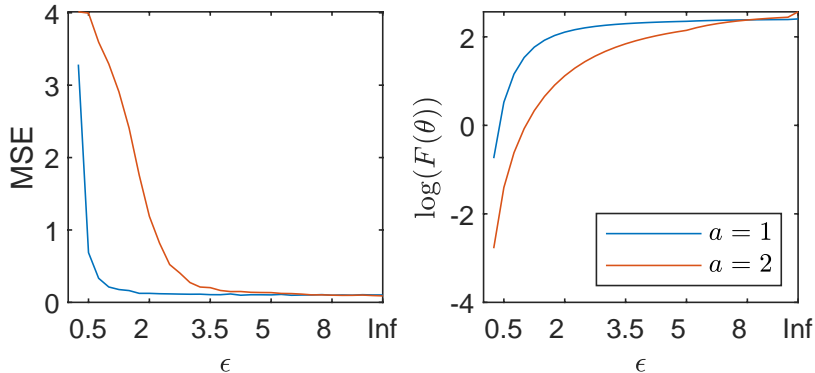
Figure 6: MSE for Algorithm 4 and (Logarithm of) $F(\theta)$ for different moments when there is Gaussian noise.

5.2 Comparison of additive statistics with the Laplace mechanism

In this part, we repeat the previous experiment but with the following differences: We consider the Laplace mechanism, where the additive statistic is corrupted by Laplace noise as

$$Y = \frac{1}{n} \sum_{i=1}^{n} |x_i|^a + V, \quad V \sim \text{Laplace}(0, A^a/(n\epsilon)),$$

For Bayesian inference, we used Algorithm 5. Note that one could use Algorithm 6 as well, which would yield the same qualitative results in terms of MSE.

Figure 7 shows MSE, obtained with $M = 100$ noisy observations, and $F(\theta)$ for the choices $s(x) = |x|$ and $s(x) = x^2$. We observe that, like in the case where we use Gaussian mechanism, $s(x) = |x|$ provides more information than $s(x) = x^2$ under the Laplace noise. Moreover, MSE values and $F(\theta)$ are consistent also in this problem.
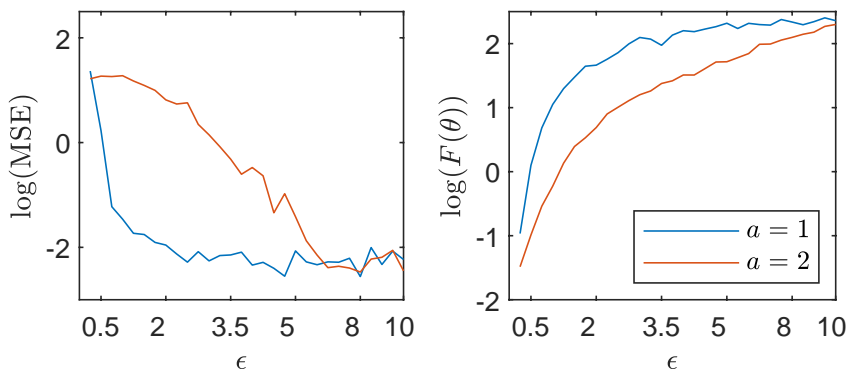

Figure 7: MSE (left) and $F(\theta)$ (right) for $s(x) = |x|$ (blue) and $s(x) = x^2$ (red), under Laplace mechanism. MSE is calculated from the samples obtained from Algorithm 5.

5.3 Comparison of Algorithms 5 and 6 in terms of mixing

Recall that Algorithms 5 and 6 are instances of PMMH and MHAAR, respectively, that both target the same posterior distribution in Section 4.2. In this part we compare their performances in terms of integrated auto-correlation (IAC) time for θ, which is the asymptotic variance of an average of samples generated by the MCMC algorithm relative to that of the average of i.i.d samples from the target distribution. (Hence, smaller IAC time is preferable.)

We continue with the setting in Section 5.2. We compared the IAC times of the algorithms with $s(x) = |x|$ and $\epsilon = 5$.

For Algorithm 5, the importance sampling distribution for u was selected as $q_\theta(u) = f_{S_n}(u|\theta)$. For Algorithm 6, we chose the symmetric proposal distribution for u as $q_{\theta,\theta'}(u) = f_{S_n}(u|(\theta + \theta')/2)$. In both algorithms, we used the same flat prior and the same random walk proposal for θ.

Table 3 shows for both algorithms the IAC times vs sample size N to estimate the acceptance ratios. As can be seen from Table 3, Algorithm 6 outperforms Algorithm 5 with lower IAC values for almost all of the N's, while the performance gap closes as N increases.

Table 3: IAC values of Algorithms 5 and 6

N	Algorithm 5	Algorithm 6
2	44.03	17.99
5	28.19	17.10
10	21.11	16.13
20	18.16	15.44
50	15.32	13.78
100	16.42	15.86

5.4 Inference based on a non-additive statistic

In this part we demonstrate the use of statistic selection method as well as the inference method when the compared statistics $S_n(X_{1:n})$ are non-additive. Specifically, we choose the maximum of $s(x_i)$'s

$$S_n(X_{1:n}) = \max\{s(X_i); i = 1, \ldots, n\}, \tag{21}$$

and the median of $s(x_i)$'s

$$S_n(X_{1:n}) = \text{median}\{s(X_i); i = 1, \ldots, n\} \tag{22}$$

as two competitors for the statistic to be shared privately. As discussed in Section 3.3, adding noise to the maximum and median based on the global sensitivity is ineffective, because the global sensitivity of the those functions are determined by the range of $s(\cdot)$ irrespective of n. Instead, we consider generating the noisy statistic Y by adjusting the amount of noise using the smooth sensitivity of the maximum and median functions.

The smooth sensitivity formulas for the maximum and median can be found in Nissim et al. (2007), we give them here for the sake of completeness. Let $A_s = \max_{x \in \mathcal{X}} s(x)$ and assume $\min_{x \in \mathcal{X}} s(x) = 0$. (Otherwise $s(\cdot)$ can be shifted by a constant so that the minimum of their range is 0.) Given the function $s(\cdot)$ and $x_1, \ldots, x_n$, let $s_1, \ldots, s_n$ be the sorted values of $s(x_1), \ldots, s(x_n)$ so that $0 \leq s_1 \leq s_2 \leq \ldots \leq s_n \leq A_s$. For the maximum in (21), the smooth sensitivity is given by

$$\Delta_{\max,\beta}^{smooth}(x_{1:n}) = \max\{e^{-k\beta} b_k; k = 0, \ldots, n\},$$

with $b_k = \max\{A_s - s_{n-k}, s_n - s_{n-k-1}\}$. For the median in in (22), the smooth sensitivity is

$$\Delta_{\mathrm{med},\beta}^{smooth}(x_{1:n}) = \max\{e^{-k\beta} b_k; k = 0, \ldots, n\}$$

with $b_k = \max\{s_{m+i} - s_{m+i-k-1}; i = 0, \ldots, k+1\}$.

As in the previous examples, we have the same population distribution, $\mathcal{P}_\theta = \mathcal{N}(0, \theta)$, and the data generation process limits X's to $[-A, A]$.

We ran Algorithm 7 with each of the above choices for S_n with $s(x) = |x|$. We took $\theta = 2$, $n = 100$, and the differential privacy parameters are taken as $(\epsilon = 5, \delta = 1/n^2)$. Table 4 shows the MSE values obtained with $M = 100$. We also report in Figure 8 the estimates of $F(\theta)$ for the median and maximum statistics, obtained with Algorithm 2, for various values of θ.

Table 4: MSE for median and maximum statistics

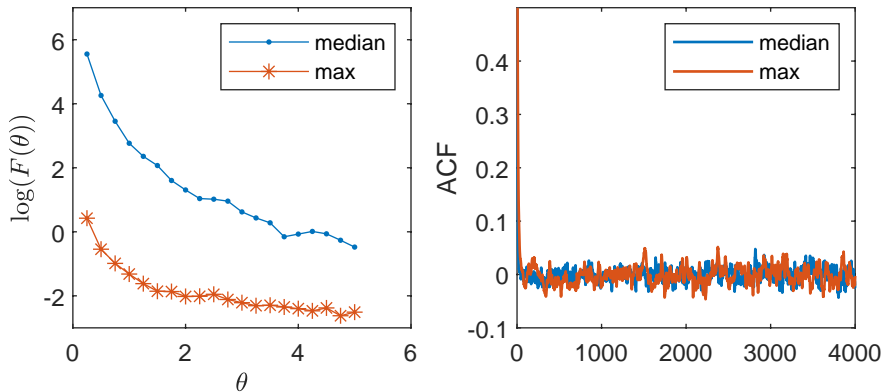| $S_n(X_{1:n})$	MSE
median	0.391
max	22.64

Figure 8: $F(\theta)$ (left) for median (blue) and maximum (red) of $s(x) = |x|$, Auto-correlation function (ACF) for Algorithm 7 for median (blue) and maximum (red). Privacy parameters are $(\epsilon, \delta) = (5, 1/n^2)$.

By observing $F(\theta)$ values and MSE values in Figure 8 and Table 4, we can see that empirical results agree with the theoretical expectations. In Figure 8, in terms of $F(\theta)$,

median has better performance since it is definitely more informative as it can be interpreted from $F(\theta)$ values. Also, MSE values reveal that estimates obtained by using median statistics are closer and variate less from the desired parameter even in the non-additive and non-gaussian case.

Figure 8 shows the sample auto-correlation function, averaged over 5 runs each with an independent noisy observation, for the median and maximum for $|x_i|$'s. We observe from the plots that Algorithm 7 mixes well for both statistics, which suggests that the MSE calculations are reliable.

5.5 Comparison of statistics in sequential release

In this part, we utilize Algorithm 8 to compare statistics using sequential release. Laplace mechanism and normal posterior distribution with unknown variance is again the target in this case. However, as it was described in Section 4.4, algorithm aims to draw samples from individual noisy data points instead of summary statistics such as mean or median.

Comparison of $s(x) = |x|$ and $s(x) = x^2$ are represented in Figure 9. We deduce from the figure that that $s(x) = |x|$ yields smaller MSE, as predicted by $F(\theta)$.
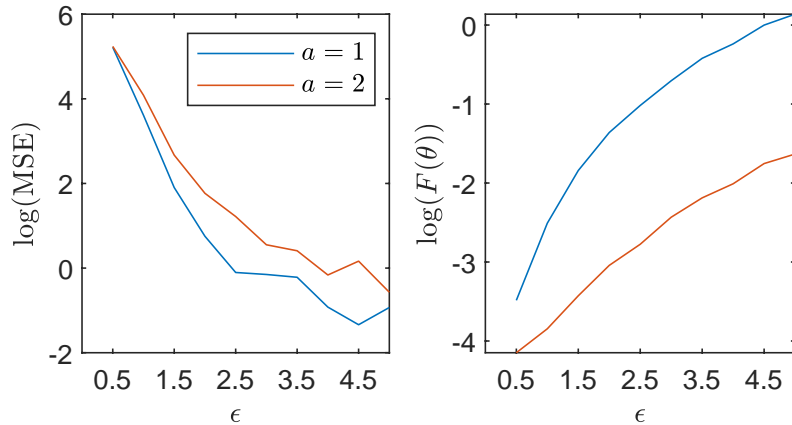

Figure 9: MSE (left) and $F(\theta)$ (right) for $s(x) = |x|$ (blue) and $s(x) = x^2$ (red), under Laplace mechanism using sequential release. MSE is calculated from the samples obtained from Algorithm 8.

6 Conclusion

In this paper, we propose a method for statistic selection for parameter estimation in a data privacy context. The method is based on the Fisher information. When one candidate statistics are not uniformly better than the other in terms of its Fisher information, the prior information for the parameter can be incorporated to make a final decision. To calculate the Fisher information, we propose several Monte Carlo algorithms for various data-sharing scenarios depending on the nature of the statistic and the privatization mechanism. We equip the statistic selection method with suitable MCMC algorithms for Bayesian parameter estimation given the shared (noisy) statistics of data. Our findings showed the usefulness

of the statistic selection based on the Fisher information as well as the effectiveness of the proposed MCMC algorithms.

The proposed framework for selecting the statistic to be privately shared is not presented as a competitor of differentially private estimation methods. In principle, the method can be useful and incorporated into any likelihood-based parameter estimation algorithm by providing the most informative statistic among those considered. Bayesian estimation via MCMC is adopted and promoted in this paper not only due to offering incorporation of the prior distribution for θ but also for the breadth of the models for which it can be applied. However, statistic selection based on Fisher information can also be utilised for differentially private maximum likelihood estimation via EM as in Gong (2019). Moreover, the work developed in this paper can be used in the schemes of Dwork and Smith (2010), where several estimators, obtained from batches, are combined into one private estimator.

The methodology presented in this paper is not specific to additive mechanisms in differential privacy. Moreover, it also extends to other definitions of privacy. Specifically, a privacy preserving mechanism can be constructed to satisfy a certain privacy level with respect to a privacy definition. The necessary condition for the presented methodology to be applied for that mechanism is the ability to write the conditional distribution of the generated output given the sensitive data.

One limitation of the work arises from the possibility of one Fisher information matrix not being greater than the other (in the sense of the difference being positive definite). In such a case, an alternative overall measure such as the trace of the Fisher information can be considered.

One possible extension of this work is adaptive clipping method in an online estimation setting where individuals' data are entered into the system sequentially and one-by-one. In such a case, each individual data can be received after clipping (so that the sensitivity is shrunk). The range of clipping can be determined in an adaptive way based on the data received so far. Adaptive clipping is already used for differentially private gradient-based algorithms (Pichapati et al., 2019; Andrew et al., 2021). It would be interesting to compare those methods to one that applies clipping to maximize informativeness of the clipped data.

Supplementary material

The code to generate the numerical results in this paper can be found at
https://github.com/barisalparslan1/Statistic_Selection_and_MCMC.

Acknowledgments

A Proof of Proposition 1

Proof of Proposition 1. We will prove the Proposition for the more general version where a subset of $z_{1:n}$ is updated.

Fix a subset $b \subseteq \{1, \ldots, n\}$. Let $z := z_{1:n}$ and $\mu(z) = \prod_{t=1}^{n} \mu(z_t)$ for a short-hand notation. Consider the joint distribution

$$\pi_b(\theta, \theta', z^{(1:N)}, k) := \eta(\theta)\mu(z^{(1)})h(y|z^{(1)}, \theta)q(\theta'|\theta) \prod_{i=2}^{N} R_b(z^{(i)}|z^{(1)}) \frac{h(y|z^{(k)}, \theta')}{\sum_{k'} h(y|z^{(k')}, \theta')}$$

where $R_b(\cdot|\cdot)$ is some conditional distribution whose selection will prove critical.

Finally, let B be the random variable corresponding to the subset b whose probability distribution is denoted by $\xi(b) = \mathbb{P}(B = b)$. Consider the extended distribution

$$\pi(\theta, \theta', z^{(1:N)}, k) = \sum_{b \subseteq \{1, \ldots, n\}} \xi(b)\pi_b(\theta, \theta', z^{(1:N)}, k)$$

The important point about $\pi(\theta, \theta', z^{(1:N)}, k)$ is that the marginal probability density of $\theta, z^{(1)}$ is the desired posterior distribution in (18) evaluated at $\theta, z^{(1)}$ and the rest of the variables are the auxiliary variables to enable a tractable MCMC algorithm. Therefore, one can sample from $\pi(\theta, \theta', z^{(1:N)}, k)$ and consider the components $\theta, z^{(1)}$, in particular the former, as samples from the true posterior distribution.

We show that when $B = b$ is sampled, Algorithm 7 targets $\pi_b(\theta', z^{(2:N)}, k|\theta, z^{(1)})$. Its proposal mechanism of corresponds to sampling $\theta', z^{(2:N)}, k$ from their conditional distribution $\pi_b(\theta', z^{(2:N)}, k|\theta, z^{(1)})$ and proposing the swapping

$$\theta \leftrightarrow \theta', \quad z^{(1)} \leftrightarrow z^{(k)}.$$

The resulting acceptance ratio is

$$\frac{\pi_b(\theta', \theta, z^{(k)}, z^{(1:k-1)}, z^{(k+1:N)}, k)}{\pi_b(\theta, \theta', z^{(1)}, \ldots, z^{(N)}, k)}$$

$$= \frac{q(\theta|\theta')\eta(\theta')\mu(z^{(k)})h(y|z^{(k)}, \theta') \prod_{i \neq k}^{N} R_b(z^{(i)}|z^{(k)}) \frac{h(y|z^{(1)}, \theta)}{\sum_{i=1}^{N} h(y|z^{(i)}, \theta)}}{q(\theta'|\theta)\eta(\theta)\mu(z^{(1)})h(y|z^{(1)}, \theta) \prod_{i=2}^{N} R_b(z^{(i)}|z^{(1)}) \frac{h(y|z^{(k)}, \theta')}{\sum_{i=1}^{N} h(y|z^{(i)}, \theta')}}$$

$$= \frac{q(\theta|\theta')\eta(\theta')\mu(z^{(k)})h(y|z^{(k)}, \theta') \prod_{i \neq k}^{N} R_b(z^{(i)}|z^{(k)}) \frac{h(y|z^{(1)}, \theta)}{\sum_{i=1}^{N} h(y|z^{(i)}, \theta)}}{q(\theta'|\theta)\eta(\theta)\mu(z^{(1)})h(y|z^{(1)}, \theta) \prod_{i=2}^{N} R_b(z^{(i)}|z^{(1)}) \frac{h_{\theta'}(y|z^{(k)})}{\sum_{i=1}^{N} h(y|z^{(i)}, \theta')}}$$

$$= \frac{q(\theta|\theta')\eta(\theta')\mu(z^{(k)}) \prod_{i \neq k}^{N} R_b(z^{(i)}|z^{(k)})}{q(\theta'|\theta)\eta(\theta)\mu(z^{(1)}) \prod_{i=2}^{N} R_b(z^{(i)}|z^{(1)})} \frac{\sum_{i=1}^{N} h(y|z^{(i)}, \theta')}{\sum_{i=1}^{N} h(y|z^{(i)}, \theta)}$$

If the distribution $\mu(z^{(1)}) \prod_{i=2}^{N} R_b(z^{(i)}|z^{(1)})$ is exchangeable with respect to $z^{(1:N)}$, then the acceptance ratio above simplifies to

$$\frac{q(\theta|\theta')\eta(\theta')}{q(\theta'|\theta)\eta(\theta)} \frac{\sum_{k'=1}^{N} h(y|z^{(k')}, \theta')}{\sum_{k'=1}^{N} h(y|z^{(k')}, \theta)}.$$

The proposal mechanism for the z variable in Algorithm 7, which corresponds to R_b here, satisfies the exchangeability property just mentioned. Hence, conditional on $B = b$, one iteration of Algorithm 7 targets $\pi_b(\theta, \theta', z^{(1:N)}, k)$.

The proof is complete by observing that one iteration of Algorithm 7 targets a $\pi_b(\theta, \theta', z^{(1:N)}, k)$ with probability $\xi(b)$, hence it targets $\pi(\theta, \theta', z^{(1:N)}, k)$. $\qquad\square$

References

Andrew, G., Thakkar, O., McMahan, H. B., and Ramaswamy, S. (2021). Differentially private learning with adaptive clipping. 6

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72:269–342. 4.2

Andrieu, C. and Roberts, G. O. (2009). The pseudo-marginal approach for efficient Monte Carlo computations. *Annals of Statistics*, 37(2):569–1078. 1, 4.2, 4.2.1, 4.4

Andrieu, C., Yıldırım, S., Doucet, A., and Chopin, N. (2020). Metropolis-hastings with averaged acceptance ratios. 1, 4.2, 4.2.2, 4.3, 4.4

Avella-Medina, M. (2019). Privacy-preserving parametric inference: a case for robust statistics. *CoRR*, abs/1911.10167. 1

Bernstein, G. and Sheldon, D. (2018). Differentially private Bayesian inference for exponential families. In *NeurIPS*. 1, 3.1

Bun, M. and Steinke, T. (2016). Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Proceedings, Part I, of the 14th International Conference on Theory of Cryptography - Volume 9985*, pages 635–658, New York, NY, USA. Springer-Verlag New York, Inc. 2, 2

Cam, L. L. (1986). *Asymptotic Methods In Statistical Decision Theory*. Springer. 3

Deligiannidis, G., Doucet, A., and Pitt, M. K. (2018). The correlated pseudomarginal method. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(5):839–870. 4.2.2

Dong, J., Roth, A., and Su, W. J. (2022). Gaussian differential privacy. *Journal of the Royal Statistical Society Series B*, 84(1):3–37. 2, 2, 3.1

Dwork, C. (2006). Differential privacy. In Bugliesi, M., Preneel, B., Sassone, V., and Wegener, I., editors, *Automata, Languages and Programming*, pages 1–12, Berlin, Heidelberg. Springer Berlin Heidelberg. 1, 1

Dwork, C. (2008). Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pages 1–19. Springer. 2

Dwork, C. and Lei, J. (2009). Differential privacy and robust statistics. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, pages 371–380, New York, NY, USA. Association for Computing Machinery. 1

Dwork, C. and Roth, A. (2013). The algorithmic foundations of differential privacy. *Theoretical Computer Science*, 9(3-4):211–407. 1, 2, 3.1

Dwork, C. and Smith, A. (2010). Differential privacy for statistics: What we know and what we want to learn. *Journal of Privacy and Confidentiality*, 1(2). 1, 6

Fearnhead, P. and Prangle, D. (2012). Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(3):419–474. 1

Foulds, J., Geumlek, J., and an Kamalika Chaudhuri, M. W. (2016). On the theory and practice of privacy-preserving Bayesian data analysis. Technical report, arxiv:1603.07294. 1

Gong, R. (2019). Exact inference with approximate computation for differentially private data via perturbations. *arXiv:1909.12237*. 1, 4, 6

Heikkilä, M. A., Jälkö, J., Dikmen, O., and Honkela, A. (2019). Differentially private Markov chain Monte Carlo. In *NeurIPS*. 1

Karwa, V., Slavković, A. B., and Krivitsky, P. (2014). Differentially private exponential random graphs. In Domingo-Ferrer, J., editor, *Privacy in Statistical Databases*, pages 143–155, Cham. Springer International Publishing. 1

Kasiviswanathan, S. P., Lee, H. K., Nissim, K., Raskhodnikova, S., and Smith, A. (2008). What can we learn privately? In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 531–540. 1, 3.4

Lei, J. (2011). Differentially private M-estimators. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc. 1

Li, B., Chen, C., Liu, H., and Carin, L. (2019). On connecting stochastic gradient MCMC and differential privacy. In Chaudhuri, K. and Sugiyama, M., editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 557–566. PMLR. 1

Nissim, K., Raskhodnikova, S., and Smith, A. (2007). Smooth sensitivity and sampling in private data analysis. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing*, STOC '07, pages 75–84, New York, NY, USA. Association for Computing Machinery. 1, 8, 5.4

Park, M., Vinaroz, M., and Jitkrittum, W. (2021). ABCDP: Approximate Bayesian computation with differential privacy. *Entropy*, 23(8). 1

Pichapati, V., Suresh, A. T., Yu, F. X., Reddi, S. J., and Kumar, S. (2019). AdaCliP: Adaptive clipping for private SGD. *ArXiv*, abs/1908.07643. 6

Räisä, O., Koskela, A., and Honkela, A. (2021). Differentially private Hamiltonian Monte Carlo. 1

Smith, A. (2011). Privacy-preserving statistical estimation with optimal convergence rates. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, STOC '11, pages 813–822, New York, NY, USA. Association for Computing Machinery. 1

Wang, Y.-X., Fienberg, S., and Smola, A. (2015). Privacy for free: Posterior sampling and stochastic gradient Monte Carlo. In Blei, D. and Bach, F., editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2493–2502. JMLR Workshop and Conference Proceedings. 1

Williams, O. and Mcsherry, F. (2010). Probabilistic inference and differential privacy. In Lafferty, J., Williams, C., Shawe-Taylor, J., Zemel, R., and Culotta, A., editors, *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc. 1

Yıldırım, S., Andrieu, C., and Doucet, A. (2018). Scalable Monte Carlo inference for state-space models. *arXiv preprint arXiv:1809.02527*. 4.4

Yıldırım, S. and Ermiş, B. (2019). Exact MCMC with differentially private moves. *Statistics and Computing*, 29(5):947–963. 1