Exact Privacy Guarantees for Markov Chain Implementations of the Exponential Mechanism with Artificial Atoms

Anonymous Author(s)
Affiliation
Address
email

Abstract

Implementations of the exponential mechanism in differential privacy often require sampling from intractable distributions. When approximate procedures like Markov chain Monte Carlo (MCMC) are used, the end result incurs costs to both privacy and accuracy. Existing work has examined these effects asymptotically, but implementable finite sample results are needed in practice so that users can specify privacy budgets in advance and implement samplers with exact privacy guarantees. In this paper, we use tools from ergodic theory and perfect simulation to design exact finite runtime sampling algorithms for the exponential mechanism by introducing an intermediate modified target distribution using artificial atoms. We propose an additional modification of this sampling algorithm that maintains its ϵ-DP guarantee and has improved runtime at the cost of some utility. We then compare these methods in scenarios where we can explicitly calculate a δ cost (as in (ϵ, δ)-DP) incurred when using standard MCMC techniques. Much as there is a well known trade-off between privacy and utility, we demonstrate that there is also a trade-off between privacy guarantees and runtime.

1 Introduction

1.1 Problem setup

The exponential mechanism [15] is one of the workhorses of ϵ-differential privacy (ϵ-DP). Many common DP mechanisms, such as the Laplace Mechanism [5], K-norm and K-norm gradient mechanisms [21, 2], pMSE mechanism [23], and even posterior sampling under certain regularity conditions [24], can all be viewed as instantiations of the exponential mechanism. In fact, any mechanism can be equivalently expressed as an instance of the exponential mechanism depending on the choice of loss function and base measure [1]. However, the output of the exponential mechanism often has an intractable distribution, making it difficult to sample from in practice. This necessitates using a sampling algorithm such as Markov chain Monte Carlo (MCMC) to computationally generate an approximate sample from the distribution. However, methods like these only approximate the target distribution asymptotically, meaning that the resulting sample need not satisfy the originally prescribed privacy guarantees.

In practice, analyzing the privacy cost incurred due to sampling is a messy problem. Existing approaches rely on analyzing convergence rates of distances between the approximating distribution and the target distribution. While these approaches have produced nice theoretical results, the analyses are usually asymptotic and lack closed form expressions for constants required to determine finite-sample incurred costs. However, for uniformly ergodic chains that admit atoms in their state space, these sampling procedures are not only more directly quantifiable but also admit finite runtime

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

algorithms for exact simulation from the target distribution. This suggests that modifying the target distribution of the exponential mechanism to include an artificial atom may eliminate the privacy cost of implementing the sampling algorithm with privacy; such modifications are the focus of this paper.

1.2 Contributions

Let $f_X(y)$ be the output density of the exponential mechanism with confidential data X and output space $y \in \mathcal{Y}$ (we will fully define this notation in the methods section). In this paper, we consider modifications to this target density as mixtures of the form $k \in [0, 1]$:

$$\tilde{f}_X(y) = (1-k)f_X(y) + kg_X(y).$$

In particular, we introduce two different methods which we will give names and briefly describe:

- `ConfAtomPerfect` $g_X(y)$ is a point mass at the confidential data output, and the target distribution is $\tilde{f}_X(y)$ conditioned on NOT releasing the confidential data, yielding an exact draw from f_X.
- `RandomAtomPerfect` $g_X(y)$ is the mass function for an implementation of the discrete exponential mechanism over a finite test point space.

Our contributions are as follows:

1. We use classical methods from MCMC theory to derive exact sampling algorithms for instances of the exponential mechanism with (ϵ, δ)-privacy guarantees.
2. We propose an exact finite-runtime algorithm (`ConfAtomPerfect`) for implementing the exponential mechanism with ϵ-DP guarantees using atomic regeneration.
3. We derive a modification of the previous algorithm (`RandomAtomPerfect`) that satisfies ϵ-DP but outputs a sample from a modified target distribution with smaller utility and shorter runtime.
4. We compare the three methods above for a few explicit examples to demonstrate a diminishing returns effect on runtimes versus privacy guarantees.

1.3 Related literature

Results from MCMC theory have addressed the problem above by analyzing convergence rates of total variational distance [17] and Renyi divergence [10]. In the latter paper, the authors argue against characterizing the distance between the MCMC sampled distribution and the target distribution in terms of total variation because it lacks the privacy-preserving compositional properties of Reyni-DP and has better dependence on the dimension of the output space. Our work differs in a few key directions. First, we require weaker assumptions about the loss function. In particular, V-geometric ergodicity only requires sub-exponential tails, which relaxes strict convexity and/or Lipschitz assumptions about the loss function. Second, we consider simpler classes of base MCMC algorithms that may be sufficient for a wide class of problems where the output dimension is smaller than the input dimension. Third, our work derives algorithms with exact implementation steps and exact privacy guarantees. Previous results state their results in terms of asymptotic sample complexities for the Langevin dynamics step size and run time, neither of which are immediately helpful for practitioners who want to use these algorithms and calculate their provable guarantees.

2 Methods

2.1 Notation

We recall some definitions and theorems here:

Definition 1 ([5]). *Let $\mathcal{M} \triangleq \{\mu_X \mid X \in \mathcal{X}^n\}$ be a collection of probability measures over a common measurable space $(\mathcal{Y}, \mathcal{F})$ indexed by elements of $\mathcal{X}^n$. The mechanism $\mathcal{M}$ satisfies (ϵ, δ)-DP if, for all $B \in \mathcal{F}$ and adjacent $X, X' \in \mathcal{X}^n$:*

$$\mu_X(B) \leq e^{\epsilon}\mu_{X'}(B) + \delta.$$

Definition 2 ([15]). *Let $L_X : \mathcal{X}^n \times \mathcal{Y} \mapsto [0, \infty]$ be a measurable loss function for each $X \in \mathcal{X}^n$. Suppose, for all adjacent $X, X' \in \mathcal{X}^n$ and $B \in \mathcal{F}$:*

$$|L_X(y) - L_{X'}(y)| \leq \Delta_L < \infty.$$

The exponential mechanism is the $(\epsilon, 0)$-DP mechanism defined by the collection of measures each with density:

$$f_X(y) \propto \exp\left(-\frac{\epsilon L_X(y)}{2\Delta_L}\right),$$

with respect to a common base measure $\nu(y)$ over $(\mathcal{Y}, \mathcal{F})$.

Frequently, ν is taken to be the Lebesgue measure over a subset of $\mathbb{R}^d$. Throughout the paper, let $\{\mu_{m,X}\}_{m=1}^{\infty}$ be a sequence of probability measures such that $\mu_{m,X}$ converges in distribution to μ_X for all $X \in \mathcal{X}^n$. We assume the chain is defined by the transition kernel $\Pi_X : \mathcal{Y} \times \mathcal{F} \mapsto [0, \infty)$ and the starting value $y_0 \in \mathcal{Y}$. We require one key assumption in our results:

Assumption 1 (Minorization Condition). *There exists a function $s : \mathcal{Y} \mapsto [0, 1]$ and probability measure ν on $(\mathcal{Y}, \mathcal{F})$ such that for all $A \in \mathcal{F}$ and $y \in \mathcal{Y}$:*

$$\Pi_X(y, A) \geq s(y)\nu(A).$$

Under this assumption, the transition kernel can be decomposed into the minorization component and the remainder component $R_{\nu,s}$:

$$\Pi_X(y, A) = s(y)\nu(A) + (1 - s(y))R_{\nu,s}(y, A).$$

This assumption will give us uniform ergodicity properties later on. Note that a limitation of our approach is that we cannot apply these methods to exponential mechanisms with unbounded output spaces such as $\mathbb{R}^k$, but it does work for compact sets of $\mathbb{R}^k$. In practice, this limitation is not a major barrier to generalizability, as the the output space is often intentionally bounded to control the sensitivity of the loss function, although there are exceptions (e.g. [23]).

The following result shows that sampling from the exponential mechanism with an asymptotic approximation from a Markov chain induces a δ penalty to the privacy budget:

Theorem 1 ([17]). *A mechanism $\mathcal{M}_m \triangleq \{\mu_{m,X} \mid x \in \mathcal{X}\}$ approximating the exponential mechanism, $\mathcal{M}$, at time $m \geq \tau(\alpha)$ is $(\epsilon, \delta_\alpha)$-DP where $\delta_\alpha \triangleq \alpha(1 + e^\epsilon)$ if*

$$\tau(\alpha) \triangleq \sup_{X \in \mathcal{X}^n} \inf\{t \geq 0 \mid \|\mu_{t,X} - \mu_X\|_{TV} \leq \alpha\}.$$

Note that the upper bound on the δ cost incurred above is specific to the sequence's properties (for example, transition probabilities and starting locations for MCMC). However, the upper bound is not specific to the confidential data X. Such generality is necessary to preserve (ϵ, δ)-DP, as the mixing time can vary based on the confidential data. This means to ensure privacy, one must find realizations of X that are "least favorable" to ensure privacy in the worst-case scenario.

One common common heuristic here is to choose $\delta \leq 1/n$, so that the relative privacy risk violation affects, in the worst case scenario, all information about one individual in the dataset. One common "fast-mixing" regime for these chains is a geometric rate, where the total variation distance is $O(r^m)$ for some $r \in (0, 1)$:

Definition 3 (V-geometric ergodicity [16]). *For a Markov chain where $\mu_{m,X} \to \mu_X$ in distribution as $m \to \infty$, we say the chain is V-geometrically ergodic if there exists constants $C \in \mathbb{R}^+$, $r \in (0, 1)$, and a function $V : \mathcal{Y} \mapsto [1, \infty)$ such that for all $m \in \mathbb{N}$ and starting points $y_0 \in \mathbb{Y}$:*

$$\|\mu_{m,X} - \mu_X\|_{TV} \leq \|\mu_{m,X} - \mu_X\|_V \leq CV(y_0)r^{-m}. \tag{1}$$

When V is a constant function, we say the chain is uniformly ergodic.

Many common algorithms, such as Metropolis-Hastings [16] and common variants such as Hybrid Monte Carlo [22] and Hamiltonian Monte Carlo [14], are geometrically ergodic under mild regularity conditions. For these algorithms, it immediately follows that for any specific realization $X \in \mathcal{X}^n$, if we choose $m = \Omega(\log(n))$:

$$\|\mu_{m,X} - \mu_X\|_{TV} \leq CV(y_0)r^{\log(n)} \implies \delta_\alpha = O(1/n). \tag{2}$$

Therefore, up to a constant, for a fast mixing sampling algorithm we require at least order $\log(n)$ samples to achieve a negligible δ for an empirical privacy guarantee. However, there are two complications to formalizing this idea. First, the constant of proportionality for the rate depends on ϵ, r, V, and C. Second, the mixing time depends on X, and the privacy guarantee depends on fast mixing for all possible $X \in \mathcal{X}^n$.

2.2 Properties of Markov chains whose state spaces admit atoms

The complications above can be alleviated when the target distribution admits an atom, specifically a proper accessible atom with respect to the transition kernel Π_X:

Definition 4 (Proper Accessible Atom []). *A set $\gamma \in \mathcal{F}$ is a proper accessible atom if:*

1. *(Regeneration measure) There exists a measure μ_{regen} on $(\mathcal{Y}, \mathcal{F})$ such that:*
$$\Pi_X(y, A) = \mu_{\text{regen}}(A) \qquad \forall y \in \gamma, A \in \mathcal{F}.$$
2. *(Accessibility criteria)*
$$\Pi_X(y, \gamma) > 0 \quad \forall y \in \mathcal{Y}.$$
3. *(Recurrence criteria) the chain returns infinitely often to γ with probability 1.*

When the state space admits an atom, the constants in Equation 1 have closed form expressions (see [16, Theorem 4.1] for an example). Furthermore, Markov chains that admit atoms have a special split chain representation as discovered by [19], where the state space is extended with a binary indicator indicating presence in the atom (i.e. $\mathcal{Y} \times \{0, 1\}$. For this split chain, let τ be the time it takes for the split Markov chain to return to said proper accessible atom after leaving it. Hobert et al. [11] show that the stationary distribution admits an infinite mixture form:

$$\mu_X(A) = \sum_{m=1}^{\infty} \frac{\mathbb{P}(\tau \geq m)}{\mathbb{E}[\tau]} \mathbb{P}(Y_m \in A \mid \tau \geq m).$$

Any algorithm can generate a perfect sample whenever it is possible to sample from the mixture components and the different conditional distribution above. Following [13], the authors focus on the singleton case when $\gamma = \{a\}$ for some $a \in \mathcal{Y}$ and the minorization measure is uniform. Define:

$$p(y) \triangleq \Pi_X(y, \{a\}).$$

By construction, for transition kernels meeting Assumption 1, $\inf_{y \in \mathcal{Y}} p(y) > \beta > 0$ for some $\beta \in \mathbb{R}$. This yields Algorithm 1 to produce an exact sample from the target distribution.

Because many target distributions do not admit the atom required, Brockwell [4] propose a way to introduce an artificial atom into a new stationary density $\tilde{f}_X$. In our setting, we propose introducing an atom at the confidential response mixed with the ϵ-DP stationary distribution with proportions $1 - k$ and k, respectively, for some $k \in (0, .5)$:

$$\tilde{f}_X(y) \triangleq (1 - k) f_X(y) + k \mathbb{1}_{\{y=a\}},$$

with respect to a new base measure:

$$\tilde{\nu} \triangleq (1 - k)\nu + k\xi_a,$$

where ν is the original base measure and ξ_a is the Dirac delta function representing a point mass at the confidential data. In particular, if the Metropolis-Hastings kernel has symmetric proposal density $q_X(y, y')$, then one can construct a new Metropolis-Hastings transition kernel $\tilde{\Pi}$ with symmetric proposal density:

$$\tilde{q}(y, y') = \frac{1}{2} \left[q_X(y, y') + \mathbb{1}_{\{y'=a\}} \right].$$

2.3 Application to private sampling

As a natural first step, we can choose to add an artificial atom at the confidential data. For our purposes, we assume that the loss function for the exponential mechanism L_x has a unique point $a \in \mathcal{Y}$ such that $L_x(a) = 0$. In other words, the confidential result uniquely minimizes the loss function up to an additive constant. This yields our first main result in Theorem 2.

Algorithm 1: Exact implementation for sampling from a singleton atom mixture [13]

Result: $Y \sim \tilde{f}_X$
Input: Transition kernel Π_X, remainder kernel $R_{a,p}$, singleton atom $a \in \mathcal{Y}$, minorization constant $\beta > 0$
Sample $M \sim \text{Geometric}(\beta)$, set $Y_1 = a$;
for $m = 2$ *to* M **do**
  Sample:

$$Z_m \sim \text{Bernoulli}\left(\frac{1 - p(Y_{m-1})}{1 - \beta}\right)$$

  if $Z_m = 1$ **then**
    Sample $Y_m^* \sim R_{a,p}(Y_{m-1}, \cdot)$ using rejection:
      1. Propose $Y_m^* \sim \Pi_X(Y_{m-1}, \cdot)$.
      2. Accept if $Y_m^* \neq a$, else go back to 1.
  else
    $Y_m = a$
  end
  Release Y_M.
end

Theorem 2 (`ConfAtomPerfect`)**.** *Suppose that $\mathcal{Y}$ is a compact subset of $\mathbb{R}^d$ and let L_X be a loss function based on confidential data $x \in \mathcal{X}^n$ where $L_X(a) = 0$ for a unique $a \in \mathcal{Y}$. Let Q be a Metropolis-Hastings transition kernel with symmetric proposals, and let $\tilde{Q}$ and $\tilde{q}$ be as defined above. Then:*

1. *There exists a constant $\beta > 0$ such that:*

$$\inf_{y \in \mathcal{Y}} \tilde{\Pi}_X(y, \{a\}) > \beta.$$

2. *There exists an algorithm (implemented by Algorithm 1) to sample from density f_X that satisfies ϵ-DP with expected number of total proposed samples N_{prop}:*

$$\mathbb{E}[N_{\text{prop}}] \leq \frac{48}{k^2(1-k)^2 \inf_{y \in \mathcal{Y}} p_{\text{Accept}}(y)},$$

where:

$$p_{\text{Accept}}(y) \triangleq \int_{\mathcal{Y}} q_X(y, y') \min\left\{1, \frac{f_X(y')}{f_X(y)}\right\} d\nu(y').$$

Proof. (Sketch; complete proof in appendix) By construction, the Metropolis-Hastings transition kernel is maximized at a for the modified chain. Taking $\beta \triangleq \frac{k}{2}$ yields the first result.

For the second result, the runtime is split into four components:

$$\begin{cases} N_{\text{Outer}} \triangleq \text{Number of outer loops in Algorithm 1} \\ N_{\text{Inner}} \triangleq \text{Number of rejection proposals to sample from } Y_m^* \sim R_{a,p}(Y_{m-1}, \cdot) \\ N_{\text{Bern}} \triangleq \text{Number of Bernoulli factory samples to select } Z_m \text{ in Algorithm 1} \\ N_{\text{Nonatommic}} \triangleq \text{Number of samples from } \tilde{f}_X \text{ required to sample from } f_X \end{cases}$$

In the complete proof, we show:

$$\begin{cases} \mathbb{E}[N_{\text{Outer}}] = \frac{2}{k} \\ \mathbb{E}[N_{\text{Inner}}] \leq ((1-k) \inf_{y \in \mathcal{Y}} p_{\text{Accept}}(y))^{-1} \\ \mathbb{E}[N_{\text{Bern}}] \leq \frac{24}{k} \\ \mathbb{E}[N_{\text{Nonatommic}}] = (1-k)^{-1} \end{cases}$$

This yields the desired bound. □

Algorithm 2: `ConfAtomPerfect`: ϵ-DP exact sample from exponential mechanism

Result: $Y \sim f_X$
Input: Sample space $\mathcal{Y}$ and loss function L_x satisfying the conditions of Theorem 2
while TRUE **do**
  Sample $\tilde{Y} \sim \tilde{f}_x$ using Algorithm 1. **if** $\tilde{Y} \neq a$ **then**
    Release $\tilde{Y}$.
  end
end

We make a couple remarks about this result. First, we note that the bound in the runtime can depend on the confidential data and does not require us to take the infimum over all possible data sets. While this is a nice property, it also assumes that the runtime of the algorithm is not a possible source of side channel information (more on this in the discussion). Second, the result above only uses symmetric proposals so that the criteria for transitioning to the confidential data does not depend on the proposal distribution. However, the result above can easily be extended to arbitrary proposals in cases where the minorizing constant can still be calculated.

Algorithm 2 assumes a deterministic location of the artificial atom, allowing for the exact same utility as the original exponential mechanism. However, the location of the atom can alternatively be randomly sampled from a discrete exponential mechanism implemented over a countable subset of the state space. This yields a less computationally expensive method, avoiding the additional $(1-k)^{-1}$ factor in implementing the mechanism with pure ϵ-DP, at the expense of some loss in utility due to representing the state space with a potentially less useful atom.

Corollary 1 (`RandomAtomPerfect`)**.** *In the same setup as Theorem 2, let $\{y^{(l)}\}_{l=1}^{\ell} \subset \mathcal{Y}$ and let g_X be the mass density of the exponential mechanism implemented on $\{y^{(l)}\}_{l=1}^{\ell}$. Then there exists a sampling algorithm (implemented by Algorithm 3) with target density:*

$$\tilde{f}_X(y) = (1-k)f_X(y) + kg_X(y), \tag{3}$$

with respect to the mixture measure:

$$\tilde{\eta} = (1-k)\nu + \frac{k}{\ell}\sum_{l=1}^{\ell}\xi_l,$$

with expected number of total proposed samples N_{prop}:

$$\mathbb{E}[N_{\text{prop}}] \leq \frac{48}{k^2(1-k)\inf_{y\in\mathcal{Y}} p_{\text{Accept}}(y)},$$

And utility:

$$\mu_X(S_\varepsilon) \leq \frac{1-k}{\nu(S_\varepsilon/2)}\exp\left(-\frac{\epsilon\varepsilon}{4\Delta_L}\right) + k\exp\left(-\frac{\epsilon}{2\Delta_L}\left(\varepsilon - \frac{2\Delta_L}{\epsilon}\log(\ell)\right)\right),$$

Where $S_\varepsilon \triangleq \{y \in \mathcal{Y} \mid L_X(y) \geq \varepsilon\}$.

3 Examples

3.1 d-dimensional mean with L_1 loss: comparing ϵ-DP and (ϵ, δ)-DP guarantees

As a first example, let $\mathcal{Y} = [0,1]^d$, $L_X(y) \triangleq \left\|y - \overline{X}\right\|_1$, and ν be a uniform measure over $\mathcal{Y}$. This is analogous to the multivariate Laplace mechanism but with bounded output space matching the bounded inputs we expect from $\mathcal{X}$. Note that although we can implement exact samplers in this scenario, the goal of this example is to illustrate the effect of proposal choice on runtime and privacy costs using MCMC versus using Algorithm 2. Derivation of all example results are in the appendix.

We consider two different proposals for MCMC. For the first algorithm, we derive a Metropolis-Hastings Markov chain constructed with proposal independently drawn uniformly over $[0,1]^d$. The slowest-mixing chain is easily identifiable:

$$\|\mu_{X,m} - \mu_X\|_{TV} \leq (1 - \beta_{\text{MCMC,unif}})^m,$$

Algorithm 3: `RandomAtomPerfect`: ϵ-DP sample from mixture of discrete and continuous exponential mechanisms

Result: $Y \sim \tilde{f}_X$ as in Equation 3
Input: Sample space $\mathcal{Y}$ and loss function L_x satisfying the conditions of Corollary 1
Sample $\tilde{Y}$ using Algorithm 1.
if $\tilde{Y} \neq a$ **then**
  Release $\tilde{Y}$.
else
  Release $Y \sim g_X$
end

where,

$$\beta_{\text{MCMC,Unif}} \triangleq \left(\frac{2d}{\epsilon n}(1 - e^{-\epsilon n/2d})\right)^d.$$

Derivation of this result is in the appendix. This allows us to immediately construct (ϵ, δ)-DP implementations of the exponential mechanism.

Similarly, we can use Laplace proposals of the form:

$$q_X(y, y') \propto \exp\left(-\alpha \left\|y - y'\right\|_{L_1}\right),$$

for which we have a new rate of convergence given analogously by:

$$\beta_{\text{MCMC,Lap}} \triangleq (2\alpha)^d \exp\left(-\left(\alpha d + \frac{\epsilon n}{2}\right)\right)\left(\frac{1}{\alpha}(1 - e^{-\alpha})\right)^d.$$

The runtime for Algorithm 2 depends on the confidential data. To compare the two approaches, we simulate $\overline{X}$ from three different Beta distributions: one uniform ($\overline{X} \sim \text{Beta}(1, 1)$), one with unimodal concentration ($\overline{X} \sim \text{Beta}(10, 10)$), and one with concentration at the boundary ($\overline{X} \sim \text{Beta}(.1, .1)$). All simulations are implemented in R [20] and figures were produced using ggplot2 [25] under GNU GPL v3.

The runtime results from the two algorithms are presented in Figure 1, illustrating a couple important phenomena. First, for the MCMC implementations, recall that the worst-case mixing time dictates the incurred δ cost. Even though we expect the Laplace proposals to mix faster in general, since the proposals match the form of the target distribution, the worst case mixing time is actually better for the independent uniform proposals. However, the relationship we expect is reflected in the runtime for the perfect sampling algorithms which depend on the confidential data. Second, we notice that the runtime for the perfect sampler decreases when the data is more likely to be concentrated at the center of the output space (as in $\text{Beta}(10, 10)$) as opposed to the boundary of the space (as in $\text{Beta}(.1, .1)$).

3.2 Ridge regression K-norm gradient mechanism: comparing confidential and random atoms

Next, we consider Ridge regression [9]. We will use the K-norm gradient mechanism following [21], a variant of the original exponential mechanism whose noise is asymptotically negligible as $n \to \infty$. In the interest of keeping consistent notation, let:

$$\mathcal{Y} \triangleq \{y \in \mathbb{R}^p \mid \|y\|_1 \leq B\}.$$

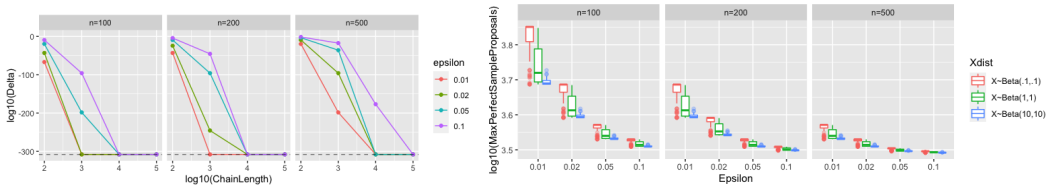
Let $X \in [-1, 1]^{n \times p}$ and $Z \in [-1, 1]^n$. Define the original loss function $\ell_X(y)$:

$$\ell_X(y) = \frac{1}{2}\|Z - Xy\|_2^2 + \frac{\lambda}{2}\|y\|_2^2.$$

For any two X, X' that differ on one element and all $y \in \mathcal{Y}$:

$$\|\nabla \ell_X(y) - \nabla \ell_{X'}(y)\|_1 \leq 2(1 + B) \sup_{x \in [0,1]^p} \|x\|_1 \leq 2(1 + B)p$$

a) Independent uniform proposals:



b) Symmetric Laplace proposals, $\alpha = \epsilon n/2$:

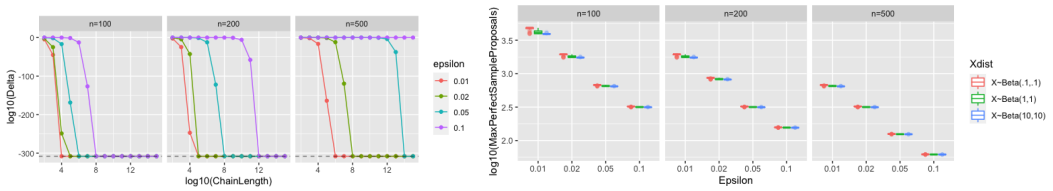


Figure 1: (Left column) delta costs incurred due to MCMC implementation with uniform independent proposals and symmetric Laplace proposals, respectively. Dashed line corresponds to smallest 64-bit double floating point precision; note this does NOT mean the resulting algorithm is ϵ-DP, but instead that the δ incurred is smaller than the smallest floating-point value reportable by the machine. (Right column) expected worst-case runtimes due to Algorithm 2 with uniform independent proposals and symmetric Laplace proposals, respectively.

This yields a final mechanism given by:

$$f_X(y) \propto \exp\left(-\frac{\epsilon}{4(1+B)p}\left\|X^T(Xy - Z) + \lambda y\right\|_1\right)\mathbb{1}_{\{\|y\|_1 \leq B\}}. \tag{4}$$

Because of similarities between functional forms, the exact sampling procedures will behave the same as the analogues in the previous example. In particular, note that the worst-case runtime calculation does not depend on the integration constant in Equation 4 when the proposals q_X are symmetric, since:

$$p_{\text{Accept}}(y) = \int_{\|y\|_1 \leq B} q_X(y, y') \min\left\{1, \frac{\exp\left(-\frac{\epsilon}{4(1+B)p}\left\|X^T(Xy' - Z) + \lambda y'\right\|_1\right)}{\exp\left(-\frac{\epsilon}{4(1+B)p}\left\|X^T(Xy - Z) + \lambda y\right\|_1\right)}\right\} d\nu(y').$$

This property is not shared by the MCMC estimator, demonstrating an advantage of this procedure.

Our main goal is to compare the utilities of Algorithms 2 and 3, meaning we need to quantify the utility cost of discretizing the output space with some probability k. To analyze this, we simulate data from a Ridge regression model with $p = 5$ and randomly sample ℓ points from the L_1 ball of radius $B = 1$ on which to implement the discrete exponential mechanism. By varying ℓ and ϵ and repeating the experiment $N_{\text{exp}} = 1000$ times, we calculate:

$$\text{Err}(\epsilon, \ell, \varepsilon) \triangleq Q_{.05}\{P(L_X(y)^{(n_{\text{exp}},\epsilon,\ell)} \geq \varepsilon)\}.$$

where $Q_{.05}$ is the 5% quantile across experiment replications. This captures with approximately 95% confidence a kind of "worst case" utility bounds by choosing a particular discretization at random for our Ridge regression. The results of the experiment are in Figure 2 with full simulation details in the appendix. Since the discrete exponential mechanism utility is a Monte Carlo estimate of the continuous exponential mechanism utility, we see as expected that the utility increases to the continuous upper bound as $\ell \to \infty$. For our particular example, this suggests that we could choose ℓ large enough that the utility of the discrete mechanism is close to that of the original exponential mechanism with high probability. However, further investigation is needed to understand these effects in higher dimensions or for more complicated loss functions.

4 Discussion

We proposed some techniques for implementing the exponential mechanism with exact privacy guarantees and some pros and cons of the various approaches above. We hope that techniques like

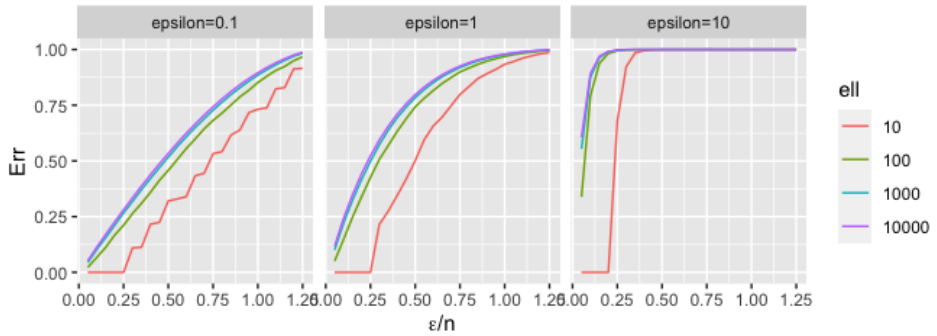


Figure 2: Utility of the discrete exponential mechanism using ℓ points sampled uniformly from the support of the distribution in Equation 4.

these can be considered when practitioners are either unwilling to incur a δ cost or are otherwise worried about the leakage of sensitive information due to disclosure of implementation details.

4.1 Limitations

As with any real-valued private computation, we must be careful about the complications of floating-point arithmetic and other side channel vulnerabilities [18]. Just because a machine cannot report a δ cost below its floating point precision does not mean the system is invulnerable. In particular, runtime issues pose a practical threat to privacy preservation when using these mechanisms; longer runtimes tend to correspond to rarer events, leading to indirect leakage. This implies our methods are best used for non-interactive data analyses, in which runtime information leakage is a lesser concern. Future work could address methods of masking or privatizing the total runtime in a way that masks how much computation time went into sampling from the target distribution.

Additionally, we have restricted our examples to relatively well-behaved distributions that allow us to analytically characterize the constants needed for exact sampling. Numerical analysis of these constants are not the primary focus of the literature on perfect sampling, but understanding additional practical applications to estimate or otherwise bound these constants would be worth investigating. Additionally, the methods we propose suffer from curse of dimensionality problems and may become prohibitively expensive for large d as noted in [10].

Finally, we reiterate that uniform ergodicity of the original chains is not only sufficient but often necessary for the existence of perfect sampling methods with finite expected runtime [13]. This requirement is often enabled by restrictions on either the confidential data space or the output of the mechanism. However, this may not be desirable in practice, and some mechanisms such as [23] do not require the same restrictions.

4.2 Future research

First, understanding the utility trade-off between Algorithms 2 and 3 requires a deeper understanding of generic utility for the exponential mechanism. Although there has been some of work on the asymptotic utility of the exponential mechanism such as [1, 8], surprisingly little work has been done on exact finite sample guarantees. In the special case when the loss function is convex, the exponential mechanism's target density is log-concave, yielding many nice properties in terms of information concentration [3]. These should be further investigated to determine when modifications to the target distribution like those proposed here are prohibitive to utility in realized terms.

Additionally, the upper bound provided in Theorem 1 may be loose in practice, and alternative convergence diagnostics besides total variation distance may be used to better quantify the privacy guarantees up to an approximation (see [7] for a comparison of some Monte Carlo error estimation approaches). We expect there to be cases where Langevin dynamics or other sampling techniques with desirable mixing properties in practice could quickly produce mathematically negligible δ values; the only open question is turning this heuristic into a quantifiable guarantee.

References

[1] Jordan Awan, Ana Kenney, Matthew Reimherr, and Aleksandra Slavković. Benefits and pitfalls of the exponential mechanism with applications to hilbert spaces and functional pca. In *International Conference on Machine Learning*, pages 374–384. PMLR, 2019.

[2] Jordan Awan and Aleksandra Slavković. Structure and sensitivity in differential privacy: Comparing k-norm mechanisms. *Journal of the American Statistical Association*, pages 1–20, 2020.

[3] Sergey Bobkov, Mokshay Madiman, et al. Concentration of the information in data with log-concave distributions. *Annals of Probability*, 39(4):1528–1543, 2011.

[4] Anthony E Brockwell and Joseph B Kadane. Identification of regeneration times in mcmc simulation, with application to adaptive schemes. *Journal of Computational and Graphical Statistics*, 14(2):436–458, 2005.

[5] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[6] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[7] James M Flegal, Murali Haran, and Galin L Jones. Markov chain monte carlo: Can we trust the third significant figure? *Statistical Science*, pages 250–260, 2008.

[8] James Foulds, Joseph Geumlek, Max Welling, and Kamalika Chaudhuri. On the theory and practice of privacy-preserving bayesian data analysis. *arXiv preprint arXiv:1603.07294*, 2016.

[9] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[10] Arun Ganesh and Kunal Talwar. Faster differentially private samplers via rényi divergence analysis of discretized langevin mcmc. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 7222–7233. Curran Associates, Inc., 2020.

[11] James P Hobert, Christian P Robert, et al. A mixture representation of π with applications in markov chain monte carlo and perfect sampling. *The Annals of Applied Probability*, 14(3):1295–1305, 2004.

[12] Mark Huber. Nearly optimal bernoulli factories for linear functions. *Combinatorics, Probability and Computing*, 25(4):577–591, 2016.

[13] Anthony Lee, Arnaud Doucet, and Krzysztof Łatuszyński. Perfect simulation using atomic regeneration with application to sequential monte carlo. *arXiv preprint arXiv:1407.5770*, 2014.

[14] Samuel Livingstone, Michael Betancourt, Simon Byrne, Mark Girolami, et al. On the geometric ergodicity of hamiltonian monte carlo. *Bernoulli*, 25(4A):3109–3138, 2019.

[15] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pages 94–103. IEEE, 2007.

[16] K L Mengersen and R L Tweedie. Rates of Convergence of the Hastings and. *The Annals of Statistics*, 24(1):101–121, 1996.

[17] Kentaro Minami, HItomi Arai, Issei Sato, and Hiroshi Nakagawa. Differential privacy without sensitivity. In *Advances in Neural Information Processing Systems*, pages 956–964, 2016.

[18] Ilya Mironov. On significance of the least significant bits for differential privacy. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 650–661, 2012.

[19] Esa Nummelin. A splitting technique for Harris recurrent Markov chains. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 43(4):309–318, 1978.

[20] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2021.

[21] Matthew Reimherr and Jordan Awan. Kng: The k-norm gradient mechanism. *arXiv preprint arXiv:1905.09436*, 2019.

[22] Gareth Roberts, Jeffrey Rosenthal, et al. Geometric ergodicity and hybrid markov chains. *Electronic Communications in Probability*, 2:13–25, 1997.

[23] Joshua Snoke and Aleksandra Slavković. pMSE mechanism: differentially private synthetic data with maximal distributional similarity. In *International Conference on Privacy in Statistical Databases*, pages 138–159. Springer, 2018.

[24] Yu-Xiang Wang, Stephen Fienberg, and Alex Smola. Privacy for free: Posterior sampling and stochastic gradient monte carlo. In *International Conference on Machine Learning*, pages 2493–2502. PMLR, 2015.

[25] Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 4.1
   (c) Did you discuss any potential negative societal impacts of your work? [No] This paper describes techniques that alleviate an existing flaw in applying privacy-preserving methods, which would improve their security in practice. Although there are potential negative impacts of privacy-preserving analysis, such as with fairness and the ability to study small populations, our paper does not exacerbate any already existing issues.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Assumption 1 and the setup for Theorem 2 and Corollary 1.
   (b) Did you include complete proofs of all theoretical results? [Yes] Sketches are in main document; complete proofs are in supplement.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code is in the supplement.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Code is in the supplement.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] Figures 1 and 2 are primarily for illustrative purposes, and the error bars at our sample sizes are negligible across simulations.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] Figures 1 and 2 are primarily for illustrative purposes, and generating the figures does not require external computing resources.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] Cited R and ggplot2
   (b) Did you mention the license of the assets? [Yes] All assets are under GNU GPL v3
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Code for the figures is in the supplement.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]