Mirror Descent based Database Privacy

Prateek Jain[1] and Abhradeep Thakurta[2]

[1] Microsoft Research India, prajain@microsoft.com
[2] Pennsylvania State University, azg161@cse.psu.edu

Abstract. In this paper, we focus on the problem of private database release in the interactive setting: a trusted database curator receives queries in an online manner for which it needs to respond with accurate but privacy preserving answers. To this end, we generalize the IDC (*Iterative Database Construction*) framework of [15,13] that maintains a differentially private artificial dataset and answers incoming *linear* queries using the artificial dataset. In particular, we formulate a generic IDC framework based on the Mirror Descent algorithm, a popular convex optimization algorithm [1]. We then present two concrete applications, namely, cut queries over a bipartite graph and linear queries over low-rank matrices, and provide significantly tighter error bounds than the ones by [15,13].

1 Introduction

Statistical analysis is extensively used to mine interesting information/patterns from the data. However, releasing such information can potentially compromise privacy of the individual records in the data [8,11,4], hence risk leaking sensitive information, e.g., health/financial records of a person/company.

Existing literature on privacy preserving statistical analysis studies the problem in two different settings: *interactive* and *non-interactive*. In the interactive setting, a database curator who owns a dataset (e.g. a hospital/bank) tries to answer queries about the dataset accurately (i.e., with small error), while preserving privacy of each individual in the dataset. In the non-interactive setting, the curator releases a "sanitized" version of the dataset that accurately answers all the queries in a given query class [2,9]. While non-interactive setting has been extensively explored in the literature[2,18,14,9,12], interactive-setting is relatively less-explored with most results being fairly recent [19,15,13].

In this paper, we focus on the interactive setting mentioned above, where the queries can be adaptively (and even adversarially) chosen according to past queries and their responses. For privacy, we use the notion of *differential privacy* [7,6] which is one of the most successful and theoretically sound notions, and is now being accepted as a benchmark. Intuitively, the output of a differentially private algorithm running on a dataset should be almost independent of the inclusion (or exclusion) of any individual data record. It is trivial to achieve privacy by giving a response that is completely independent of the underlying data. However, such a response will have large error and hence low *utility*.

A slightly better solution is to independently add enough noise to each query response such that it nullifies the effect of any particular record in the dataset. However, to preserve privacy for k queries with this scheme, naïve analysis suggests that the error in each query scales as $O(\sqrt{k})$. In the pursuit of obtaining a better error bound than $O(\sqrt{k})$, [19] proposed an algorithm called the *median mechanism* that improves over the naïve solution, and guarantees $O(\frac{\text{poly}(\log k)}{N^{1/3}})$ error in each query response for *adaptive* queries over normalized histograms. Here N is the number of records in the database. While their result reduced the dependence on the number of queries to $\log k$, the error bound was still higher than the sampling error of $1/\sqrt{N}$. Furthermore, in general the algorithm is super-polynomial in both N and k.

Recently, [15] proposed a multiplicative weights update (MW) based algorithm that can guarantee $O(\sqrt{(\log k)/N})$ error for *linear queries* over normalized histograms. Their method maintains a differentially private artificial dataset at each step. For a given query, if the existing artificial dataset provides an answer close to the true response then the artificial dataset is not updated. Otherwise, the artificial dataset is updated so that it gets "closer" to the true dataset. [15] show that a multiplicative update to the dataset requires a small number of updates and hence only a small amount of noise needs to be added at each step.

Subsequently, [13] proposed a more generic framework which, given an update mechanism or Iterative Database Construction scheme (IDC) for maintaining artificial dataset, can guarantee privacy as well as *utility* (i.e., bound on the error in each query response). Utility guarantee by [13] depends on the number of updates that the given IDC might require in the worst case. Moreover, [13] also proposed an update scheme based on Frieze/Kannan (FK) cut decomposition method and provided utility guarantee for the same.

In this paper, we use the framework of [13] and provide a generic Iterative Database Construction scheme (IDC) based on the Mirror Descent algorithm, a popular convex optimization method [1]. Our Mirror Descent based IDC (MD-IDC) scheme can be adapted for any strongly convex potential function. Further, we provide a bound on the number of updates required by our MD-IDC scheme and thus obtain privacy and utility guarantees using framework of [13]. We show that the MW update based IDC (MW-IDC) and the FK algorithm based IDC (FK-IDC) are special cases of our generic MD-IDC and their utility guarantees follows directly from our generalized analysis.

Depending on the structure of the set of queries as well as the geometry of the dataset, MD-IDC can provide different utility guarantees for different potential functions. We provide examples where, by selecting different potential function than the ones used by [15,13], we can obtain tighter error bounds.

Next, we apply our framework to the problem of releasing cut values in a bipartite graph and propose an algorithm that guarantees smaller error than both [15] and [13]. For this problem, we use a *group*-norm based potential function that is known to exploit sparsity structure in the data [22]. Similarly, we apply our framework to the problem of releasing linear queries over a dataset that is a low-rank matrix. We show that by using spectral structure of both the underlying

matrix and the queries, our method guarantees smaller error than the methods of [13] and [15].

Our Contributions:

1. **Unify and generalize MW-IDC and FK-IDC [15,13]:** We propose a generic Mirror Descent based IDC (MD-IDC) which is a generalized update rule from which MW-IDC and FK-IDC can be derived as special cases.
2. **Exploit geometry of true dataset and queries:** Using our Mirror Descent-IDC, we can capture a wider class of structural properties on the underlying dataset $\boldsymbol{x}^*$ and the set of linear queries $\mathcal{F}$. Specifically, we provide potential functions for MD-IDC that can directly exploit bound on any arbitrary L_p-norm of $\boldsymbol{x}^*$ and the L_q-norm over $\mathcal{F}$. In contrast, [15,13] are limited to (L_1, L_∞) and (L_2, L_2) norm pair, respectively.
3. **Application to graph cuts release and linear query release over low-rank matrices:** We compare our utility bounds against the ones provided by MW-IDC and FK-IDC on three practically relevant applications: i) interactive cut query release for *imbalanced* bi-partite graphs (i.e., bi-partite graphs with large degree variations), ii) interactive cut query release for *power-law distributed* bipartite graphs (i.e., bipartite graphs where degree distribution follows a power-law), and iii) private matrix sensing where goal is to release responses to linear queries over a *low-rank* matrix. In each case, we show that our error bounds are significantly tighter than the ones obtained by the existing methods [15,13].

Paper Outline: In Section 3, we formulate our problem and discuss the framework of [13] in Section 3.1. Then, in Section 3.2, we propose our Mirror Descent based IDC algorithm and provide utility guarantees for the same. In Section 4, we provide two applications of our MD-IDC framework, namely, 1) releasing cut-queries in bi-partite graph, 2) releasing linear queries over low-rank matrices.

2 Notation and Preliminaries

Let $\boldsymbol{x}^* \in \mathbb{R}^d$ denote the private dataset, $\mathcal{F} = \{f_1, \ldots, f_k\}$ denote the function sequence provided to the online query response algorithm. For every query function $f \in \mathcal{F}$, we assume $f : \mathbb{R}^d \to \mathbb{R}$ to be a linear function, denoted by $f(\boldsymbol{x}) = \langle f, \boldsymbol{x} \rangle$. Vectors are denoted by bold-face symbols (e.g., $\boldsymbol{x}$), matrices are represented by capital letters (e.g., M). X_i denotes i-th row of X. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ denotes the inner product between vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. Similarly, $\mathrm{Tr}(X^T Y) = \langle X, Y \rangle$ denotes the inner product between X and Y. L_p norm or p-norm of a vector $\boldsymbol{x} \in \mathbb{R}^d$ is denoted as $\|\boldsymbol{x}\|_p = \left(\sum_i^d x_i^p\right)^{1/p}$ and $p^* = \frac{p}{p-1}$ denotes the dual norm of L_p. For a matrix X, $\|X\|_p$ represents L_p-norm of vectorized X. $\|X\|_F = \sqrt{\sum_{ij} X_{ij}^2}$ denotes the Frobenius norm of X.

Definition 1 (**(p,q)-group norm of matrix X**) $\|X\|_{p,q} = (\sum_{i=1}^m \|X_i\|_p^q)^{1/q}$, *where $X \in \mathbb{R}^{m \times n}$. Hence, (p,q)-group norm is equivalent to L_q norm of a vector of L_p norms of rows of X.*

Definition 2 *Let $X = U\Sigma V^T$ be the singular value decomposition of X. Then Schatten p-norm of X is given by: $\|X\|_{S_p} = (\sum_i \sigma_i^p)^{1/p}$, where σ_i is the i-th singular value of X and $\sigma_1 \geq \sigma_2 \ldots$.*

Definition 3 (Uniform convexity) *A function $\Psi : \mathbb{R}^d \to \mathbb{R}$ is s-uniformly convex (for $s \geq 1$) with respect to $\|\cdot\|_r$ iff: $\forall \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d, \forall \alpha \in [0,1], \Psi_r(\alpha \boldsymbol{x} + (1-\alpha)\boldsymbol{y}) \leq \alpha \Psi_r(\boldsymbol{x}) + (1-\alpha)\Psi_r(\boldsymbol{y}) - \frac{\alpha(1-\alpha)}{s}\|\boldsymbol{x} - \boldsymbol{y}\|_r^s$*

Note that the definition above is a generalization of the conventional strong convexity definition where s is set to be two.

Definition 4 *Let $\Psi : \mathbb{R}^d \to \mathbb{R}$ be a continuously differentiable strictly convex potential function. Then, the Bregman's divergence (generated by Ψ) between any two vectors $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^d$ is defined as:*

$$\Delta_\Psi(\boldsymbol{x}_1; \boldsymbol{x}_2) = \Psi(\boldsymbol{x}_1) - \Psi(\boldsymbol{x}_2) - \langle \bigtriangledown \Psi(\boldsymbol{x}_2), \boldsymbol{x}_1 - \boldsymbol{x}_2 \rangle.$$

3 Problem Definition and Overview

Given a private dataset $\boldsymbol{x}^* \in \mathbb{R}^d$ and a set of queries $\mathcal{F} = \{f_1, \ldots, f_i \ldots f_k\}, f_i : \mathbb{R}^d \to \mathbb{R}, \forall i$, the goal is to answer each query f_i accurately (w.r.t $\boldsymbol{x}^*$) while preserving privacy of $\boldsymbol{x}^*$. That is, if a_i is the response to query f_i, then we want:

$$|a_i - f_i(\boldsymbol{x}^*)| \leq T, \forall 1 \leq i \leq k,$$

while preserving privacy of $\boldsymbol{x}^*$; $T > 0$ is an error parameter.

The above mentioned problem is known as the *interactive dataset release* problem [19,15]. In this setting, the queries can be adversarial, that is the adversary can select f_i depending on responses to previous queries. Hence, the privacy of each response a_i has to be argued w.r.t. complete query set $\mathcal{F}$.

For privacy, we use the notion of differential privacy which is now a benchmark notion [7,6]. Intuitively, an algorithm is differential private if addition (removal) of an entry to (from) the dataset does not significantly alter the output. In the context of *interactive dataset release*, it requires a guarantee that *none* of the query response a_i change significantly, if one entry of the dataset $\boldsymbol{x}^*$ is modified. Below, we provide a formal definition of $(\epsilon, \delta, \gamma)$-differential privacy adapted for the problem of interactive dataset release.

Definition 5 (Differential privacy [7,6]) *An algorithm $\mathcal{A}$ is $(\epsilon, \delta, \gamma)$-differentially private if for any two datasets $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^d$ s.t. $\|\boldsymbol{x} - \boldsymbol{x}'\|_1 \leq \gamma$, and for all measurable sets $\mathcal{O} \subseteq Range(\mathcal{A})$, the following holds:*

$$\Pr[\mathcal{A}(\boldsymbol{x}) \in \mathcal{O}] \leq e^\epsilon \Pr[\mathcal{A}(\boldsymbol{x}') \in \mathcal{O}] + \delta.$$

Now, a special case of the above mentioned problem is when each query f_i is linear, i.e., $f_i(\boldsymbol{x}) = \langle f_i, \boldsymbol{x} \rangle$, $f_i \in \mathbb{R}^d$. Most of the existing results are for the case of linear queries only. For rest of the paper, we assume f_i to be a linear query; we discuss extension to the nonlinear case in the full version of this paper [16].

Algorithm 1 Online Query Response Mechanism (OQR) [15,13]

Require: Dataset: $\boldsymbol{x}^*$, privacy parameters: $(\epsilon, \delta, \gamma)$, query set $\mathcal{F} = \{f_1, \cdots, f_k\}$, failure probability β,
U_{IDC}: IDC algorithm, B: bound on number of updates by U_{IDC}

Set noise parameter: $\epsilon_0 \leftarrow \frac{\epsilon}{100\gamma\sqrt{B}\log(4/\delta)}$, Set threshold $T \leftarrow \frac{4}{\epsilon_0}\log(2k/\beta)$
$\boldsymbol{x}_0 = U_{IDC}(\text{NULL}, \text{NULL}, \text{NULL})$, counter $= 0$.
for $t \in \{1, \cdots, k\}$ and counter $< B$ **do**
  $A_t \sim \text{Lap}(\frac{1}{\epsilon_0})$
  True response: $a_t = f_t(\boldsymbol{x}^*)$, Noisy response: $\hat{a}_t \leftarrow f_t(\boldsymbol{x}^*) + A_t$, Noisy difference: $\hat{d}_t \leftarrow \hat{a}_t - f_t(\boldsymbol{x}_{t-1})$
  if $|\hat{d}_t| > T$ **then**
    $\boldsymbol{x}_t \leftarrow U_{IDC}(\boldsymbol{x}_{t-1}, f_t, \hat{d}_t)$, counter $\leftarrow$ counter $+ 1$
    Output query response: $\hat{a}_t = f_t(\boldsymbol{x}^*) + A_t$
  else
    No update, i.e., $\boldsymbol{x}_t \leftarrow \boldsymbol{x}_{t-1}$
    Output query response: $\hat{a}_t = f_t(\boldsymbol{x}_t)$
  end if
end for

Recently, [15] provided a multiplicative weights update based differentially private algorithm for the problem of *interactive dataset release* (with linear queries) that guarantees at most $O(\log^{1/4} k \log d)$ error in each query. Subsequently, [13] proposed a more general framework that uses *Iterative Database Construction (IDC)* algorithms to provide differentially private versions of dataset $\boldsymbol{x}^*$. [13] provided a tighter analysis of the multiplicative weights based algorithm (MW-IDC) of [15]. They also proposed a novel IDC algorithm based on Frieze/Kannan cut-decomposition algorithm (FK-IDC) [10] and apply their method to the problem of releasing graph cuts.

In the next section, we introduce the above mentioned *online query release mechanism* of [13] and state the generic utility and privacy guarantee of [13]. Then, in section 3.2, we present our generic Mirror Descent based IDC (MD-IDC) and show that both MW-IDC and FK-IDC form special cases of our MD-IDC algorithm. Further, their error bounds follow directly from our generic analysis for MD-IDC. We also provide two applications where different instantiations of our MD-IDC provide better error bounds than MW-IDC and FK-IDC.

3.1 Online Query Release Mechanism

see Step 5

[15,13] introduced a generic online query release mechanism where at each step t, a differentially private (or "public") version of the dataset $\boldsymbol{x}_{t-1}$ is maintained. Now, for a given query f_t (that can be adversarially chosen according to $\boldsymbol{x}_{t-1}$ and past query responses), the algorithm tries to answer the query using $\boldsymbol{x}_{t-1}$. However, if query response $f_t(\boldsymbol{x}_{t-1})$ is "too far" from the true response $f_t(\boldsymbol{x}^*)$, then the algorithm answers the query based on the true dataset $\boldsymbol{x}^*$. Also,

as the dataset $\boldsymbol{x}_{t-1}$ is "inaccurate", hence it is *updated* so that it gets closer to $\boldsymbol{x}^*$. The *update* algorithm is called *Iterative Database Construction (IDC)* algorithm, and should produce next iterate $\boldsymbol{x}_t$ using the previous iterate $\boldsymbol{x}_{t-1}$, current query f_t, and the response provided for f_t. That is, $U_{IDC} : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$, where U_{IDC} is the given IDC algorithm. See Algorithm 1 for a pseudo-code.

Now, [15] observed that, for iterations where iterate $\boldsymbol{x}_{t-1}$ is not updated, Algorithm 1 is ($\epsilon = 0$)-differentially private with high probability over the randomness of the algorithm. Also, $\boldsymbol{x}_{t-1}$ is updated only for a small number of steps. Using these observations, [15,13] show that the noise parameter set in Step 1 of Algorithm 1 is enough to guarantee privacy of $\boldsymbol{x}^*$.

Theorem 1 (Privacy (Theorem 4.1, [13])) *Assuming each query $f \in \mathcal{F}$, $\|f\|_\infty \leq 1$, Algorithm 1 is $(\epsilon, \delta, \gamma)$-differentially private.*

Similar to [15,13], utility (i.e., maximum error in any query response) can be guaranteed easily by bounding the magnitude of the noise added using tail bounds for Laplace distribution.

Theorem 2 (Utility) *If the variable counter (defined in Step 2 of Algorithm 1 (Algorithm* OQR *)) is less than B after all the k-query responses, then with probability $\geq 1 - \frac{\beta}{2}$, Algorithm* OQR *incurs at most $2T$ error in each query response, i.e.,*

$$|\hat{a}_t - f_t(\boldsymbol{x}^*)| \leq 2T = \frac{800\gamma\sqrt{B}\log(4/\delta)\log(2k/\beta)}{\epsilon}, \forall 1 \leq t \leq k,$$

where B is the bound on number of updates using U_{IDC}.

Note that privacy guarantee of Algorithm 1 is independent of the IDC algorithm (U_{IDC}), while the utility guarantee depends on U_{IDC} only through a bound on the number of updates (B). Hence, the most critical aspect of Algorithm 1 is the design of U_{IDC} and provide a tight upper bound on B for the given application. In next section, we present a generic Mirror Descent algorithm based IDC algorithm that can be adapted according to the underlying application to obtain better bound on B (and hence the utility guarantee).

3.2 Mirror Descent based IDC

In this section, we introduce our Mirror Descent based IDC. Mirror descent is a popular optimization algorithm [1], that is also extensively used in the context of online learning [20]. Suppose, the goal is to minimize a function $\ell(\boldsymbol{x})$ s.t. $\boldsymbol{x} \in \mathcal{C}$ where $\mathcal{C}$ is a convex set. Then, mirror descent uses the following *exploration-exploitation* based update (with $\Delta(\cdot;\cdot)$ being the distance function):

$$\boldsymbol{x}_t = \arg\min_{\boldsymbol{x} \in \mathcal{C}} \left(\Delta(\boldsymbol{x}; \boldsymbol{x}_{t-1}) - \eta_t \langle \bigtriangledown \ell(\boldsymbol{x}_{t-1}), \boldsymbol{x} \rangle\right). \tag{1}$$

For online query release mechanism (Algorithm OQR (Algorithm 1)), we use a similar MD-based update to design IDC. Specifically, we set $\ell_t(\boldsymbol{x}) = |f_t(\boldsymbol{x}^*) -$

Algorithm 2 Mirror Descent based IDC (MD-IDC)

Require: Previous iterate: $\boldsymbol{x}_{t-1}$, Linear query: $f_t \in \mathcal{F}$, Norm parameters: p, q, Noisy difference in response: $\hat{d}_t = \langle f_t, \boldsymbol{x}^* \rangle + A_t - \langle f_t, \boldsymbol{x}_{t-1} \rangle$, Threshold: T, Privacy parameters: $(\epsilon, \delta, \gamma)$, $\zeta_q = \max_{f \in \mathcal{F}} \|f\|_q$, Potential function: Ψ that is s-uniformly convex w.r.t. $\|\cdot\|_r$, $r = \frac{q}{q-1}$

1: Define $\mathcal{C} = \{\boldsymbol{x} \; s.t. \; \|\boldsymbol{x}\|_p \leq \|\boldsymbol{x}^*\|_p\}$

2: Set step size $\eta = \frac{(s-1)^{s-1}(T/2)^{s-1}}{s^s \zeta_q^s}$ and update bound $B = \frac{2^{s-1} s^s \zeta_q^s}{T^s (s-1)^{s-1}} \max_{\boldsymbol{x} \in \mathcal{C}} \Psi(\boldsymbol{x})$

3: **if** $\boldsymbol{x}_{t-1} = \phi$ (i.e., $t = 1$) **then**

4: Output: $\boldsymbol{x}_0 = \operatorname{argmin}_{\boldsymbol{x} \in \mathcal{C}} \Psi(\boldsymbol{x})$

5: **else**

6: Output: $\boldsymbol{x}_t \leftarrow \operatorname{argmin}_{\boldsymbol{x} \in \mathcal{C}} \left(\Delta_\Psi(\boldsymbol{x}; \boldsymbol{x}_{t-1}) - \eta \cdot \operatorname{sgn}(\hat{d}_t) \langle \bigtriangledown f_t(\boldsymbol{x}_{t-1}), \boldsymbol{x} - \boldsymbol{x}_{t-1} \rangle \right)$

7: **end if**

$f_t(\boldsymbol{x})|$. Note that, we want to update $\boldsymbol{x}_{t-1}$ so that $\ell_t(\boldsymbol{x})$ is small, i.e., $\boldsymbol{x}_t$ does not make mistakes on queries similar to f_t. But at the same time, we want $\boldsymbol{x}_t$ to be close to $\boldsymbol{x}_{t-1}$, as it contains information learned from previous queries.

As $\ell_t(\boldsymbol{x}) = |f_t(\boldsymbol{x}^*) - f_t(\boldsymbol{x})|$ is not a differentiable function, we use the following sub-gradient of ℓ_t: $\partial \ell_t(\boldsymbol{x}_{t-1}) = -\operatorname{sgn}(f_t(\boldsymbol{x}^*) - f_t(\boldsymbol{x}_{t-1})) \bigtriangledown f_t(\boldsymbol{x}_{t-1})$. Also, as each function f_t is linear, i.e., $f_t(\boldsymbol{x}) = \langle f_t, \boldsymbol{x} \rangle$, $f_t \in \mathbb{R}^d$: $\bigtriangledown f_t(\boldsymbol{x}) = f_t$.

Finally, we use Bregman's divergence as the distance function $\Delta(\cdot;\cdot)$. Given a continuously differentiable strictly convex function Ψ, the corresponding Bregman's divergence is given by: $\Delta_\Psi(\boldsymbol{x}_1; \boldsymbol{x}_2) = \Psi(\boldsymbol{x}_1) - \Psi(\boldsymbol{x}_2) - \langle \bigtriangledown \Psi(\boldsymbol{x}_2), \boldsymbol{x}_1 - \boldsymbol{x}_2 \rangle$.

Hence, for a given potential function Ψ and $d_t = f_t(\boldsymbol{x}^*) - f_t(\boldsymbol{x}_{t-1})$, our MD-IDC update for linear queries is given by:

$$\boldsymbol{x}_t = \operatorname*{argmin}_{\boldsymbol{x} \in \mathcal{C}} \left(\Delta_\Psi(\boldsymbol{x}; \boldsymbol{x}_{t-1}) - \eta \cdot \operatorname{sgn}(d_t) \langle f_t, \boldsymbol{x} - \boldsymbol{x}_{t-1} \rangle \right),$$

where η is selected appropriately. See Algorithm 2 for a pseudo-code of our MD-IDC algorithm. In the following, we provide the utility guarantees for our MD-IDC based Online Query Response Mechanism (Algorithm 1).

Theorem 3 (Utility) *Let $f_t \in \mathcal{F}, 1 \leq t \leq k$ be a linear query, let q be the norm chosen for the query set $\mathcal{F}$ and let $\mathcal{C} = \{\boldsymbol{x} \; s.t. \; \|\boldsymbol{x}\|_p \leq \|\boldsymbol{x}^*\|_p\}$. Furthermore, let $\Psi(\cdot)$ be a s-strongly convex function w.r.t. $\|\cdot\|_r$, where $r = \frac{q}{q-1}$. Then, w.p. at least $1 - \beta$, for each query response, the error incurred by MD-IDC (Algorithm 2) based* OQR *algorithm (Algorithm 1) is bounded by:*

$$|\hat{a}_t - f_t(\boldsymbol{x}^*)| = O\left(\frac{\log(k/\beta)^{2/(s+2)} (\gamma \zeta_q)^{s/(s+2)} \log^2(1/\delta)}{\epsilon^{s/(s+2)}} \left(\max_{\boldsymbol{x} \in \mathcal{C}} \Psi(\boldsymbol{x}) \right)^{1/(s+2)} \right),$$

where, $1 \leq t \leq k$, $\zeta_q \leq \max_{f \in \mathcal{F}} \|f\|_q$ and $(\epsilon, \delta, \gamma)$ are the privacy parameters.

A detailed proof of the above theorem is provided in the full version [16].

Special Cases: MW-IDC & FK-IDC: Above we described our generic MD-IDC algorithm which given any *potential function*, provides bound on the error in

each query's response. Our algorithm has the flexibility of selecting the potential function for different problem settings.Recall that the potential function should be strongly convex w.r.t. $\|\cdot\|_{\frac{q}{q-1}}$-norm over set $\mathcal{C} = \{\boldsymbol{x}\ s.t.\ \|\boldsymbol{x}\|_p \leq \|\boldsymbol{x}^*\|_p\}$, while $\max_{\boldsymbol{x}\in\mathcal{C}} \Psi(\boldsymbol{x})$ should be small. Note that, here we assume that $\|\boldsymbol{x}^*\|_p$ is is known *publicly* or an approximate version of the same can released in differentially private manner, by adding appropriate amount of noise.

Now, it is known that for $1 < p \leq 2$, $\Psi_p(\boldsymbol{x}) = \frac{1}{p-1}\|\boldsymbol{x}\|_p^2$ is 2-uniformly convex w.r.t. $\|\cdot\|_p$. Selecting $p = q^*$ (i.e., p, q are *dual* pairs) and ignoring privacy parameters (ϵ, δ) and failure probability β, we get the following error bound: $\mathsf{Err}_p = O(\sqrt{\gamma\|f\|_{p^*}\|\boldsymbol{x}^*\|_p \log k})$. Now, if $\boldsymbol{x}^*$ is a histogram over a database with N records, then $\gamma = \frac{1}{N}$. Hence, $\mathsf{Err}_p = O(\sqrt{\frac{1}{N}\|f\|_{p^*}\|\boldsymbol{x}^*\|_p \log k})$.

Interestingly, selecting $p = 2$, our MD-IDC reduces to Frieze/Kannan IDC (FK-IDC) of [13]. Further, the error bound is also *exactly* the same as the one obtained by [13]. Similarly, selecting $p = \frac{\log d}{\log d - 1}$, we get the matching error bound for MW-IDC [13]. However, the algorithm is different than that of MW-IDC and is in fact more general, as it can be applied to any real-valued $\boldsymbol{x}^*$, while MW-IDC applies to positive vectors only. Further, selecting $\Psi_H(\boldsymbol{x}) = \sum_i x_i \log x_i$, we obtain exact MW-IDC algorithm. Note that, $\Psi_H(\boldsymbol{x})$ is 2-uniformly convex w.r.t. L_1 norm and hence can be applied directly in our framework.

Above, we assume p and q to be dual pairs, i.e., $q = p^*$. However, similar to [20], selecting non-dual (p, q) pair can lead to tighter bounds for certain settings. We defer the details for non-dual (p, q) pairs to the full version of the paper [16].

4 Applications

In this section, we discuss some of the applications of our MD-IDC, and show that by selecting an appropriate potential function Ψ for a given application, we can obtain significantly more accurate answers than [13,15]. In particular, we provide two concrete applications and show that we can devise problem specific potential functions to outperform the existing methods of [13,15].

4.1 Online Cut-query Release

In this section, we consider the problem of releasing cut-queries over a private *bi-partite* graph. Specifically, let $G = (V_1, V_2, E)$ be an undirected bi-partite graph and let $S \subseteq V_2$ be a subset of nodes. The goal here is to release cut $(S, \bar{S})$ while preserving privacy. The cut query answers the following question: how "well-connected" are the nodes of V_1 are to $S \subseteq V_2$. For simplicity of exposition we assume $S \subseteq V_2$; for $S \subseteq V_2 \cup V_1$, similar results can be obtained easily.

For online cut-query release, the "dataset" is given by the adjacency matrix of G, i.e., $X^* \in \mathbb{R}^{|V_1|\times|V_2|}$. $X^*_{ij} = 1, \forall (i,j) \in E, 1 \leq i \leq |V_1|, 1 \leq j \leq |V_2|$ and is zero otherwise. Similarly, a cut query is given by $F \in \mathbb{R}^{|V_1|\times|V_2|}$, where $F_{ij} = 1, \forall i \in S, j \in \bar{S}$. Hence, the cut size is given by $C(S, G) = \langle X^*, F\rangle$.

Note that, we want to guarantee privacy for each edge in the graph. Hence, removing or adding an edge from X leads to an "adjacent" dataset X'. Also, $\gamma = \|X - X'\|_1 \leq 1$. We seek an algorithm that answers queries accurately while providing $(\epsilon, \delta, \gamma = 1)$-differential privacy. For this problem, we use Algorithm 1 (Algorithm OQR) with our generic MD-IDC.

For MW-IDC [15], using Theorem 3 and $k = O(|V_2|^{|S|})$, (ignoring privacy parameters (ϵ, δ) and failure probability β) the error in each query is given by:

$$\mathsf{Err}_{MW} = O\left(\sqrt{\zeta_\infty |E||S| \log(|V_1||V_2|)}\right), \quad \zeta_\infty = \max_t \|F_t\|_\infty = 1. \tag{2}$$

Now, for FK-IDC [13], Theorem 3 provides the following bound:

$$\mathsf{Err}_{FK} = O\left(\sqrt{\zeta_2 |E|^{1/2} |S| \log(|V_2|)}\right), \quad \zeta_2 = \max_t \|F_t\|_2. \tag{3}$$

Similar to the previous section, we can select a different L_p-norm potential function for our MD-IDC, than the one used by MW-IDC, FK-IDC. However, for this problem, that does not lead to an improvement over MW-IDC and FK-IDC. Instead, with the intent of exploiting the structure of the adjacency matrix, we select group-norm based potential functions (see Definition 1). Of particular interest is the $(2, p)$-norm, where $p \approx 1$. Similar to L_p norms, it can be shown that $\Psi_{2,p}(X) = \frac{1}{p-1}\|\boldsymbol{x}\|_{2,p}^2$ is 2-uniformly convex w.r.t. $\|\cdot\|_{2,p}, 1 < p \leq 2$. Note that, this function is same as the "Group Lasso" regularizer [22] and is known to be useful for recovering vectors with shared sparsity. For our problem, this function is useful for the case where degrees of nodes in the graph have heavy variation.

Using Theorem 3, error incurred by MD-IDC with $(2, p)$-norm function is:

$$\mathsf{Err}_{MD-IDC} = O\left(\sqrt{\zeta_{2,p^*} \|X^*\|_{2,p} |S| \log(|V_2|)}\right), \tag{4}$$

where $\zeta_{2,p^*} = \max_t \|F_t\|_{2,p^*}$ and $p^* = p/(p-1)$. Note that, the error bound for our group-norm based MD-IDC is in general incomparable to the corresponding bounds by MW-IDC or FK-IDC. However for several specific problems, group-norm based MD-IDC outperforms both MW-IDC and FK-IDC. Below, we provide two such examples.

Imbalanced Bi-partite Graph: Consider a bi-partite graph where the node sets V_1 and V_2 are of equal cardinality, i.e., $|V_1| = |V_2| = V$. Let V_1 be divided into two sets $V_1 = \{A, B\}$. Let $|A| = |V|^{3/4}$ and let each node of A be connected to every node of V_2, while each node of B is connected to only $|V|^{1/2}$ nodes of B. That is a small number of nodes are highly connected, while the remaining nodes are sparsely connected. Recall that the cut-queries are over a set $S \subseteq V_2$.

Note, that for the above mentioned family of graph $|E| = O(|V|^{7/4})$. Hence, bounds for MW-IDC and FK-IDC are given by:

$$\mathsf{Err}_{MW} = \tilde{O}(|V|^{7/8}|S|^{1/2}), \quad \mathsf{Err}_{FK} = \tilde{O}(|V|^{11/16}|S|^{3/4}) \tag{5}$$

Similarly, the error incurred by $(2, p = \frac{\log|V|}{\log|V|-1})$-norm based MD-IDC is:

$$\mathsf{Err}_{2,\frac{\log|V|}{\log|V|-1}} = \tilde{O}(|V|^{5/8}|S|^{3/4}).$$

Hence, if $|S| = o(|V|)$, then:

$$\mathsf{Err}_{2,\frac{\log|V|}{\log|V|-1}} = o(1)\mathsf{Err}_{MW}, \quad \mathsf{Err}_{2,\frac{\log|V|}{\log|V|-1}} = o(1)\mathsf{Err}_{FK}.$$

Also, note that the error incurred by a trivial response of 0 for each query is bounded by: $|V||S|$. Similarly, standard randomized response leads to $O(|V|^{3/2})$ error. Hence, our error guarantees are better than the trivial baselines as well.

Power-law Distributed Bi-partite Graph: Next, we consider a more practical scenario where degrees of nodes in V_1 follow a power-law distribution. Several graphs that arise in practice have been shown to follow a power-law distribution. For simplicity, we assume $|V_1| = |V_2| = |V|$. Now, power-law distribution assumption implies: $\mathbb{E}[\text{Number of nodes with degree } i] = \frac{i^{-\beta}}{\sum_{i=j}^{|V|} j^{-\beta}}|V|$, where $\beta > 0$ is a parameter of the distribution. For simplicity, we drop expectation from the above statement and assume the following *deterministic* statement:

$$\text{Number of nodes with degree } i = \frac{i^{-\beta}}{\sum_{i=j}^{|V|} j^{-\beta}}|V|.$$

If $1 < \beta < 2$, it can be shown that: $|E| = O(|V|^{3-\beta})$. Hence, using (2), (3):

$$\mathsf{Err}_{MW} = \tilde{O}(|V|^{3/2-\beta/2}|S|^{1/2}), \quad \mathsf{Err}_{FK} = \tilde{O}(|V|^{1-\beta/4}|S|^{3/4}).$$

Similarly, using (4), for $1 < \beta < 3/2$: $\mathsf{Err}_{MD-IDC} = \tilde{O}(|V|^{3/4-\beta/2}|S|^{3/4})$. Hence, using the fact that $|S| \leq |V|$ and assuming $1 < \beta < 3/2$:

$$\mathsf{Err}_{2,\frac{\log|V|}{\log|V|-1}} = o(1)\mathsf{Err}_{MW}, \quad \mathsf{Err}_{2,\frac{\log|V|}{\log|V|-1}} = o(1)\mathsf{Err}_{FK}.$$

Finally, we compare the above mentioned error bounds with the error incurred by a trivial response of 0. For this trivial response, the error is bounded by: $\min\{|S||V|, |E|\} = \min\{|S||V|, |V|^{3-\beta}\}$. Hence, if $|S| \geq |V|^{1-\frac{2}{3}\beta}$, then $\mathsf{Err}_{2,\frac{\log V}{\log V-1}}$ is smaller than the error incurred by the trivial response. Similarly, randomized response incurs $O(|V|^{3/2})$ error. Hence, if $|S| = o(|V|)$, then our proposed MD-IDC obtains better error bounds.

Finally, we note that while our results are for online cut-queries, they can also be used for releasing *sanitized* differentially private graphs which are accurate for cut queries. However, our algorithm would require to process all $O(|V|^{|S|})$ cut-queries. We leave further investigation of our MD-IDC method for release of sanitized differentially-private graphs as future work.

4.2 Online Query Release over Low-rank Matrix

In this section, we consider the problem of releasing response to linear queries where the dataset is a low-rank matrix. Let $X^* \in \mathbb{R}^{m\times n}$ be a rank-r matrix and let $F_t \in \mathbb{R}^{m\times n}$ be a linear query. Then, the response to the query is: $\langle F_t, X^*\rangle$.
A practical scenario: let X^* be a user-movie rating matrix, i.e., $X^*_{i,j}$ is the rating user i provides for movie j. And the queries answer questions of the form: "what is the average rating for comedy movies for users from Seattle".

We can directly apply Algorithm OQR (Algorithm 1) to release response to these queries, while providing privacy guarantees for each individual entry in X^*. Assuming $\|X^*\|_\infty = 1$, for any adjacent dataset X', $\|X^* - X'\|_1 \leq 1 = \gamma$. Recall that, $\|X\|_p$ represents L_p norm of vectorized X.

Similar to the previous section, we provide a potential function for our MD-IDC that provides better error guarantees than MW-IDC and FK-IDC. Note that, the matrix X^* can have negative entries as well, hence multiplicative weight based algorithm from [15] cannot be applied directly.

Using Theorem 3 with FK-IDC, we obtain the following error bound for answer k-queries (ignoring privacy parameters (ϵ, δ) and failure probability β):

$$\mathsf{Err}_{FK} = \tilde{O}\left(\sqrt{\zeta_2 \|X^*\|_F \log k}\right), \text{ where } \zeta_2 \leq \max_t \|F_t\|_F.$$

In the previous section, we used group-norm based potential functions as they are more well-suited for exploiting degree structure of the graphs. In this section, we use another popular class of potential functions based on the Schatten-p norm (see Definition 2) that is more well-suited to exploit the spectral structure of X^*.

Similar to L_p norm, it is known that $\Psi_{S_p}(X) = \frac{1}{p-1}\|X\|^2_{S_p}, \forall 1 < p \leq 2$, is 2-strongly convex w.r.t. $\|\cdot\|_{S_p}$ [17]. Hence using Theorem 3, the error incurred by Algorithm 1 with MD-IDC and with potential function Ψ_{S_p} is bounded by:

$$\mathsf{Err}_{S_p} = \tilde{O}\left(\sqrt{\zeta_{S_p^*} \|X^*\|_{S_p} \log(k)}\right), \tag{6}$$

where $p^* = \frac{p}{p-1}$ and $\zeta_{S_p^*} = \max_t \|F_t\|_{S_{p^*}}$. Note that, for $p = 2$, S_2 is the Frobenius norm and hence in that case, the above bound is same as Err_{FK}.

Now, for the case of low-rank matrices, Schatten-1 norm (or "trace" norm) is a popular regularization as it generally preserves the low-rank structure. Below, we show for a large class of queries using trace norm based MD-IDC indeed achieves better error bounds than both MW-IDC and FK-IDC. Specifically, let $p = \frac{\log mn}{\log mn - 1} \approx 1$. Then, using (6) and $\|X^*\|_{S_1} \leq \sqrt{r}\|X^*\|_F$ we get:

$$\mathsf{Err}_{S_1} = \tilde{O}\left(\sqrt{\sqrt{r}\zeta_{S_p^*} \|X^*\|_F \log(k)}\right), \tag{7}$$

where $p^* = \log(mn)$. Hence, if $\sqrt{r}\|F_t\|_{S_{\log mn}} < \|F_t\|_F, \forall t$, then $\mathsf{Err}_{S_1} < \mathsf{Err}_{FK}$. Now, $\|F_t\|_{S_{\log mn}} \leq e\sigma_1^{F_t}$, where $\sigma_1^{F_t}$ is the largest singular value of F_t. Similarly, $\|F_t\|_F = \sqrt{\sum_i (\sigma_i^{F_t})^2}$, where $\sigma_i^{F_t}$ is the i-th singular value of F_t.

Now, if each query F_t is a rank-1 query, then, $\sqrt{r}\|F_t\|_{S_{\log mn}} > \|F_t\|_F$ for $r > 1$. Hence, in this case, Frobenius-norm based potential function leads to tighter bounds. However, if the queries have "spread-out" spectrum, then trace-norm based potential function is more accurate.

A concrete example of such a case is when each element of F_t is sampled uniformly from a standard Gaussian, i.e., $F_t(i,j) \sim N(0,1)$. In this case, using Corollary 5.35 of [21] and assuming $m > 4n$, we get (w.h.p.): $\sqrt{n} \leq \sigma_n^{F_t} \leq \sigma_1^{F_t} \leq 3\sqrt{n}$. Hence, $r(\sigma_1^{F_t})^2 \leq 9rn \leq 9r\frac{1}{n}(\sigma_n^{F_t})^2$. That is, $\sqrt{r}\|F_t\|_{S_{\log mn}} \leq 3e\sqrt{r/n}\|F_t\|_F$. Hence, for $r = o(n)$, $\mathsf{Err}_{S_1} = o(1)\mathsf{Err}_{FK}$. In typical applications, r is a constant. Hence, Err_{S_1} is a factor of $\sqrt{n}$ *smaller than* Err_{FK}.

Note that, random queries F_t are used extensively in the domain of compressed sensing [3] and can be used to recover low-rank matrix X^* accurately. Hence, our result provides a method to recover matrix X^* approximately (with bounded error) without compromising accuracy of any single entry.

References

1. Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Oper. Res. Lett.*, 31(3):167–175, 2003.
2. Avrim Blum, Katrina Ligett, and Aaron Roth. A learning theory approach to non-interactive database privacy. In *STOC*, 2008.
3. E. J. Candes and T. Tao. Near-optimal signal recovery from random projections: universal encoding strategies? In *IEEE Transactions on Information Theory*, 2006.
4. Irit Dinur, Cynthia Dwork, and Kobbi Nissim. Revealing information while preserving privacy, full version of [5], in preparation, 2010.
5. Irit Dinur and Kobbi Nissim. Revealing information while preserving privacy. In *PODS*, pages 202–210. ACM, 2003.
6. Cynthia Dwork, Krishnaram Kenthapadi, Frank Mcsherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *In EUROCRYPT*, pages 486–503. Springer, 2006.
7. Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, 2006.
8. Cynthia Dwork, Frank McSherry, and Kunal Talwar. The price of privacy and the limits of LP decoding. In *STOC*, pages 85–94. ACM, 2007.
9. Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. Boosting and differential privacy. In *FOCS*, 2010.
10. Alan M. Frieze and Ravi Kannan. A simple algorithm for constructing szemere'di's regularity partition. In *Electr. J. Comb.*, 1999.
11. Srivatsava Ranjit Ganta, Shiva Prasad Kasiviswanathan, and Adam Smith. Composition attacks and auxiliary information in data privacy. In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 265–273. ACM, 2008.
12. Anupam Gupta, Moritz Hardt, Aaron Roth, and Jonathan Ullman. Privately releasing conjunctions and the statistical query barrier. In *STOC*, 2011.
13. Anupam Gupta, Aaron Roth, and Jonathan Ullman. Iterative constructions and private data release. *CoRR*, abs/1107.3731, 2011.
14. Moritz Hardt, Katrina Ligett, and Frank McSherry. A simple and practical algorithm for differentially private data release. *CoRR*, abs/1012.4763, 2010.
15. Moritz Hardt and Guy N. Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. In *FOCS*, 2010.
16. Prateek Jain and Abhradeep Thakurta. Mirror descent based database privacy. Technical Report NAS-TR-0159-2012, Pennsylvania State University, April 2012.
17. Sham M. Kakade, Shai Shalev-Shwartz, and Ambuj Tewari. On the duality of strong convexity and strong smoothness: Learning applications and matrix regularization. *Informal publication*, 2009.
18. Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? In *FOCS*, 2008.
19. Aaron Roth and Tim Roughgarden. Interactive privacy via the median mechanism. In *STOC*, 2010.
20. Nathan Srebro, Karthik Sridharan, and Ambuj Tewari. On the universality of online mirror descent. *CoRR*, abs/1107.4080, 2011.
21. Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *CoRR*, abs/1011.3027, 2010.
22. Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68:49–67, 2007.