Information and Inference: A Journal of the IMA (2025) **14**, iaae034
https://doi.org/10.1093/imaiai/iaae034

Differentially private low-dimensional synthetic data from high-dimensional datasets

YIYUN HE*
Department of Mathematics, University of California, Irvine, CA 92697, USA
*Corresponding author. Email: yiyunh4@uci.edu

THOMAS STROHMER
Department of Mathematics and Center of Data Science and Artificial Intelligence Research, University of California, Davis, CA 95616, USA

ROMAN VERSHYNIN
Department of Mathematics, University of California, Irvine, CA 92697, USA

AND

YIZHE ZHU
Department of Mathematics, University of Southern California, Los Angeles, CA 90089, USA

[Received on 13 February 2024; revised on 24 September 2024; accepted on 10 December 2024]

Differentially private synthetic data provide a powerful mechanism to enable data analysis while protecting sensitive information about individuals. However, when the data lie in a high-dimensional space, the accuracy of the synthetic data suffers from the curse of dimensionality. In this paper, we propose a differentially private algorithm to generate low-dimensional synthetic data efficiently from a high-dimensional dataset with a utility guarantee with respect to the Wasserstein distance. A key step of our algorithm is a private principal component analysis (PCA) procedure with a near-optimal accuracy bound that circumvents the curse of dimensionality. Unlike the standard perturbation analysis, our analysis of private PCA works without assuming the spectral gap for the covariance matrix.

Keywords: differential privacy; synthetic data; principal component analysis.

1. Introduction

As data sharing is increasingly locking horns with data privacy concerns, privacy-preserving data analysis is becoming a challenging task with far-reaching impact. Differential privacy (DP) has emerged as the gold standard for implementing privacy in various applications [20]. For instance, DP has been adopted by several technology companies [25] and has also been used in connection with the release of Census 2020 data [2]. The motivation behind the concept of differential privacy is the desire to protect an individual's data while publishing aggregate information about the database, as formalized in the following definition:

DEFINITION 1 (Differential Privacy [20]). A randomized algorithm $\mathscr{M}$ is ε-differentially private if for any pair of datasets D and D' that differ on one data (i.e. $D = D_0 \cup \{X\}$ and $D' = D_0 \cup \{X'\}$ for some dataset D_0), and any measurable subset $S \subseteq \text{range}(\mathscr{M})$, we have

$$\mathbb{P}\left\{\mathscr{M}(D) \in S\right\} \leq e^{\varepsilon}\, \mathbb{P}\left\{\mathscr{M}(D') \in S\right\},$$

where the probability is with respect to the randomness of $\mathscr{M}$.

However, utility guarantees for DP are usually provided only for a fixed, predefined set of queries. Hence, it has been frequently recommended that differential privacy may be combined with synthetic data to achieve more flexibility in private data sharing [7, 30, 57]. Synthetic datasets are generated from existing datasets and maintain the statistical properties of the original dataset. Hence, the datasets can be shared freely among investigators in academia or industry, without security and privacy concerns.

Yet, computationally efficient construction of accurate differentially private synthetic data is challenging. Most research on private synthetic data has been concerned with counting queries, range queries or k-dimensional marginals, see, e.g. [9, 10, 24, 30, 50, 52, 53]. Notable exceptions are [56, 12] and [19]. Specifically, [12] provide utility guarantees with respect to the 1-Wasserstein distance. Invoking the Kantorovich-Rubinstein duality theorem, the 1-Wasserstein distance accuracy bound ensures that all Lipschitz statistics are preserved uniformly. Given that numerous machine learning algorithms are Lipschitz [13, 39, 46, 55], this provides data analysts with a vastly increased toolbox of machine learning methods for which one can expect similar outcomes for the original and synthetic data.

For instance, for the special case of datasets living on the d-dimensional Boolean hypercube $\{0,1\}^d$ equipped with the Hamming distance, the results in [12] show that there exists an ε-DP algorithm with an expected utility loss that scales like

$$\left(\log(\varepsilon n)^{\frac{3}{2}}/(\varepsilon n)\right)^{1/d}, \tag{1.1}$$

where n is the size of the dataset. While [31] succeeded in removing the logarithmic factor in (1.1), it can be shown that the rate in (1.1) is otherwise tight. Consequently, the utility guarantees in [12] and [31] are only useful when d, the dimension of the data, is small (or if n is exponentially larger than d). In other words, we are facing the curse of dimensionality. The curse of dimensionality extends beyond challenges associated with Wasserstein distance utility guarantees. Even with a weaker accuracy requirement, the hardness result from Uhlman and Vadhan [52] shows that $n = \text{poly}(d)$ is necessary for generating DP-synthetic data in polynomial time while maintaining approximate covariance.

In [19], the authors succeeded in constructing DP synthetic data with utility bounds where d in (1.1) is replaced by $(d' + 1)$, assuming that the dataset lies in a certain d'-dimensional subspace. Their notion of dimension is similar to the Minkowski dimension, and their method is applicable beyond the linear subspace setting. However, the optimization step in their algorithm exhibits exponential time complexity in d, see [19, Section D].

This paper presents a computationally efficient algorithm that does not rely on any assumptions about the true data. We demonstrate that our approach enhances the utility bound from d to d' in (1.1) when the dataset is in a d'-dimensional affine subspace. Specifically, we derive a DP algorithm to generate low-dimensional synthetic data from a high-dimensional dataset with a utility guarantee with respect to the 1-Wasserstein distance that captures the intrinsic dimension of the data.

Our approach revolves around a private principal component analysis (PCA) procedure with a near-optimal accuracy bound that circumvents the curse of dimensionality. Different from classical

perturbation analysis [15, 23] that utilizes the Davis-Kahan theorem [17] in the literature, our accuracy analysis of private PCA works without assuming the spectral gap for the covariance matrix.

Notation. In this paper, we work with data in the Euclidean space $\mathbb{R}^d$. For convenience, the data matrix $\mathbf{X} = [X_1, \ldots, X_n] \in \mathbb{R}^{d\times n}$ also indicates the dataset $(X_1, \ldots, X_n)$. We use $\mathbf{A}$ to denote a matrix and v, X as vectors. $\|\cdot\|_F$ denotes the Frobenius norm and $\|\cdot\|$ is the operator norm of a matrix. Two sequences a_n, b_n satisfies $a_n \lesssim b_n$ if $a_n \le Cb_n$ for an absolute constant $C > 0$.

Organization of the paper. The rest of the paper is arranged as follows. In the remainder of Section 1, we present our algorithm with the main theorem for privacy and accuracy guarantees in Section 1.1, followed by a discussion. A comparison to the state of the art is given in Section 1.2. Definitions and lemmas used in the paper are provided in Section 2.

Next, we consider the Algorithm 1 step by step. Section 3 discusses private PCA and noisy projection. In Section 4, we modify synthetic data algorithms from [31] to the specific cases on the lower dimensional spaces. The precise privacy and accuracy guarantee of Algorithm 1 is summarized in Section 5. We discuss an adaptive and private choice of d' in Section 6. Finally, since the case $d' = 1$ is not covered in Theorem 1, we discuss additional results under stronger assumptions in Section 7.

1.1 *Main results*

In this paper, we use Definition 1 on data matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$. We say two data matrices $\mathbf{X}, \mathbf{X}'$ are *neighboring datasets* if $\mathbf{X}$ and $\mathbf{X}'$ differ on only one column. We follow the setting and notation in [31] as follows: let (Ω, ρ) be a metric space. Consider a dataset $\mathbf{X} = [X_1, \ldots, X_n] \in \Omega^n$. We aim to construct a computationally efficient differentially private randomized algorithm that outputs synthetic data $\mathbf{Y} = [Y_1, \ldots, Y_n] \in \Omega^m$ such that the two empirical measures

$$\mu_{\mathbf{X}} = \frac{1}{n}\sum_{i=1}^{n} \delta_{X_i} \quad \text{and} \quad \mu_{\mathbf{Y}} = \frac{1}{m}\sum_{i=1}^{m} \delta_{Y_i}$$

are close to each other. Here δ_{X_i} denotes the Dirac measure centered on X_i.

We measure the utility of the output by $\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}})$, where the expectation is taken over the randomness of the algorithm. We assume that each vector in the original dataset $\mathbf{X}$ is inside $[0,1]^d$; our goal is to generate a differentially private synthetic dataset $\mathbf{Y}$ in $[0,1]^d$, where each vector is close to a linear subspace of dimension d', and the empirical measure of $\mathbf{Y}$ is close to $\mathbf{X}$ under the 1-Wasserstein distance. We introduce Algorithm 1 as a computationally efficient algorithm for this task. It can be summarized in the following four steps:

1. Construct a private covariance matrix $\widehat{\mathbf{M}}$. The private covariance is constructed by adding a Laplacian random matrix to a centered covariance matrix $\mathbf{M}$ defined as

$$\mathbf{M} = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})(X_i - \overline{X})^{\mathsf{T}}, \quad \text{where} \quad \overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i. \tag{1.2}$$

   This step is presented in Algorithm 2.

2. Find a d'-dimensional subspace $\widehat{\mathbf{V}}_{d'}$ by taking the top d' eigenvectors of $\widehat{\mathbf{M}}$. Then, project the data onto a linear subspace. The new data obtained in this way are inside a d'-dimensional ball. This step is summarized in Algorithm 3.
3. Generate a private measure in the d'-dimensional ball centered at the origin by adapting methods in [31], where synthetic data generation algorithms were analyzed for data in the hypercube. This is summarized in Algorithms 4 and 5.
4. Add a private mean vector to shift the dataset back to a private affine subspace. Given the transformations in earlier steps, some synthetic data points might lie outside the hypercube. We then metrically project them back to the domain of the hypercube. Finally, we output the resulting dataset $\mathbf{Y}$. This is summarized in the last two parts of Algorithm 1.

Our main theorem states the privacy and accuracy guarantees of Algorithm 1.

THEOREM 1. Let $\Omega = [0,1]^d$ equipped with ℓ^∞ metric and $\mathbf{X} = [X_1, \ldots, X_n] \in \Omega^n$ be a dataset. For any $2 \le d' \le d$, Algorithm 1 outputs an ε-differentially private synthetic dataset $\mathbf{Y} = [Y_1, \ldots, Y_m] \in \Omega^m$ for some $m \ge 1$ in polynomial time such that

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \lesssim \sqrt{\sum_{i>d'} \sigma_i(\mathbf{M})} + \sqrt{\frac{d' d^{2.5}}{\varepsilon n}} + \sqrt{\frac{d}{d'}} (\varepsilon n)^{-1/d'},$$

where $\sigma_i(\mathbf{M})$ is the i-th largest eigenvalue value of $\mathbf{M}$ in (1.2).

Note that m, the size of the synthetic dataset $\mathbf{Y}$, is not necessarily equal to n since the low-dimensional synthetic data subroutine in Algorithm 1 creates noisy counts. See Section 4 for more details.

Algorithm 1 Low-dimensional Synthetic Data

Input: True data matrix $\mathbf{X} = [X_1, \ldots, X_n]$, $X_i \in [0,1]^d$, privacy parameter ε.
(Private covariance matrix) Apply Algorithm 2 to $\mathbf{X}$ with privacy parameter $\varepsilon/3$ to obtain a private covariance matrix $\widehat{\mathbf{M}}$.
(Private linear projection) Let $\overline{X}_{\text{priv}}$ denote the private mean of the true dataset. Choose a target dimension d'. Apply Algorithm 3 with privacy parameter $\varepsilon/3$ to shift and project $\mathbf{X}$ onto a private d'-dimensional linear subspace.
(Low-dimensional synthetic data) Use subroutine in Section 4 to generate $\varepsilon/3$-DP synthetic data $\mathbf{X}'$ of size m depending on $d' = 2$ or $d' \ge 3$.
(Adding the private mean vector) Shift the data back by $X_i'' = X_i' + \overline{X}_{\text{priv}}$.
(Metric projection) Define $f : \mathbb{R} \to [0,1]$ such that

$$f(x) = \begin{cases} 0 & \text{if } x < 0; \\ x & \text{if } x \in [0,1]; \\ 1 & \text{if } x > 1. \end{cases}$$

Then, for $v \in \mathbb{R}^d$, we define $f(v)$ to be the result of applying f to each coordinate of v.
Output: Synthetic data $\mathbf{Y} = [f(X_1''), \ldots, f(X_m'')]$.

Optimality. There are three terms on the right-hand side of (5.1). The first term is the error from the rank-d' approximation of the covariance matrix $\mathbf{M}$. The second term is the accuracy loss for private PCA after the perturbation from a random Laplacian matrix. The optimality of this error term remains an open question. The third term is the accuracy loss when generating synthetic data in a d'-dimensional subspace. Notably, the factor $\sqrt{d/d'}$ is optimal. This can be seen by the fact that a d'-dimensional section of the cube can be $\sqrt{d/d'}$ times larger than the low-dimensional cube $[0,1]^{d'}$ (e.g. if it is positioned diagonally). Complementarily, [12] showed the optimality of the factor $(\varepsilon n)^{-1/d'}$ for generating d'-dimensional synthetic data in $[0,1]^{d'}$. Therefore, the third term in (5.1) is necessary and optimal.

Improved accuracy. When the original dataset $\mathbf{X}$ lies in an affine d'-dimensional subspace, it implies $\sigma_i(\mathbf{M}) = 0$ for $i > d'$ and $\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \lesssim \sqrt{\frac{d'd^{2.5}}{\varepsilon n}} + \sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'}$. This is an improvement from the accuracy rate $O((\varepsilon n)^{-1/d})$ for unstructured data in $[0,1]^d$ in [12, 31] when $d \le n^{\alpha_n}$ and $d' \le \min\{\frac{d}{2}, \frac{1}{\alpha_n}\}$ for $0 < \alpha_n \le \frac{2}{7}$. For example, we can take α_n to be a constant in $(0, \frac{2}{7}]$ or $\alpha_n = \frac{1}{\log\log n}$. This improved rate overcomes the curse of high dimensionality.

Adaptive and private choices of d'. The target dimension d' is a hyperparameter in Algorithm 1. One can choose the value of d' adaptively and privately based on singular values of the private covariance matrix $\widehat{\mathbf{M}}$ in Algorithm 2 such that

$$d' := \operatorname*{arg\,min}_{2 \le k \le d} \left(\sqrt{\sum_{i>d'} \sigma_i(\widehat{\mathbf{M}})} + \sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'} \right).$$

Discussion on such choice of d' is referred to Section 6.

Low-dimensional representation of X. The synthetic dataset $\mathbf{Y}$ is close to a d'-dimensional subspace under the 1-Wasserstein distance, as shown in Proposition 6.

Running time. The *private linear projection* step in Algorithm 1 has a running time $O(d^2 n)$ using the truncated SVD [41]. The *low-dimensional synthetic data* subroutine has a running time polynomial in n for $d' \ge 3$ and linear in n when $d' = 2$ [31]. Therefore, the overall running time for Algorithm 1 is linear in n, polynomial in d when $d' = 2$ and is poly(n, d) when $d' \ge 3$. Although sub-optimal in the dependence on d' for accuracy bounds, one can also run Algorithm 1 in linear time by choosing PMM (Algorithm 4) in the subroutine for all $d' \ge 2$.

1.2 *Comparison to previous results*

Private synthetic data. Most existing work considered generating DP-synthetic datasets while minimizing the utility loss for specific queries, including counting queries [9, 22, 30], k-way marginal queries [24, 52], histogram release [1]. For a finite collection of predefined linear queries Q, [30] provided an algorithm with running time linear in $|Q|$ and utility loss grows logarithmically in $|Q|$. The sample complexity can be reduced if the queries are sparse [9, 19, 24]. Beyond finite collections of queries, [56] considered utility bound for differentiable queries, and recent works [12, 31] studied Lipschitz queries with utility bound in Wasserstein distance. [19] considered sparse Lipschitz queries with an improved accuracy rate. [6, 27, 40, 58] measure the utility of DP synthetic data by the maximum mean discrepancy (MMD) between empirical distributions of the original and synthetic datasets. This metric is different

from our chosen utility bound in Wasserstein distance. Crucially, MMD does not provide any guarantees for Lipschitz downstream tasks.

Our work provides an improved accuracy rate for low-dimensional synthetic data generation. Compared to [19], our algorithm is computationally efficient and has a better accuracy rate. Besides [19], we are unaware of any work on low-dimensional synthetic data generation from high-dimensional datasets. While methods from [12, 31] can be directly applied if the low-dimensional subspace is known, the subspace would be non-private and could reveal sensitive information about the original data. The crux of our paper is that we do not assume the low-dimensional subspace is known, and our DP synthetic data algorithm protects its privacy. This setting is closely related to the problem of privately learning the subspace of the dataset considered in [23, 49,51].

Private PCA. Private PCA is a commonly used technique for differentially private dimension reduction of the original dataset. This is achieved by introducing noise to the covariance matrix [15, 23, 32, 35, 36, 45, 60]. Instead of independent noise, the method of exponential mechanism is also extensively explored [15, 35, 38]. Another approach, known as streaming PCA [34, 47], can also be performed privately [28, 43].

The private PCA typically yields a private d'-dimensional subspace $\widehat{\mathbf{V}}_{d'}$ that approximates the top d'-dimensional subspace $\mathbf{V}_{d'}$ produced by the standard PCA. The accuracy of private PCA is usually measured by the distance between $\widehat{\mathbf{V}}_{d'}$ and $\mathbf{V}_{d'}$ [23, 29, 43, 45, 49]. To prove a utility guarantee, a common tool is the Davis-Kahan Theorem [8, 59], which assumes that the covariance matrix has a spectral gap [15, 23, 28, 35, 43]. Alternatively, using the projection error to evaluate accuracy is independent of the spectral gap [5, 38, 44]. In our implementation of private PCA, we don't treat $\widehat{\mathbf{V}}_{d'}$ as our terminal output. Instead, we project $\mathbf{X}$ onto $\widehat{\mathbf{V}}_{d'}$. Our approach directly bound the Wasserstein distance between the projected dataset and $\mathbf{X}$. This method circumvents the subspace perturbation analysis, resulting in an accuracy bound independent of the spectral gap, as outlined in Lemma 3. [49] considered a related task that takes a true dataset close to a low-dimensional linear subspace and outputs a private linear subspace. To the best of our knowledge, none of the previous work on private PCA considered low-dimensional DP synthetic data generation.

Centered covariance matrix. A common choice of the covariance matrix for PCA is $\frac{1}{n}\mathbf{X}\mathbf{X}^{\mathsf{T}}$ [14, 23, 49], which is different from the centered one defined in (1.2). The rank of $\mathbf{X}$ is the dimension of the linear subspace that the data lie in rather than that of the affine subspace. If $\mathbf{X}$ lies in a d'-dimensional affine space (not necessarily passing through the origin), centering the data shifts the affine hyperplane spanned $\mathbf{X}$ to pass through the origin. Consequently, the centered covariance matrix will have rank d', whereas the rank of $\mathbf{X}$ is $d'+1$. By reducing the dimension of the linear subspace by 1, the centering step enhances the accuracy rate from $(\varepsilon n)^{-1/(d'+1)}$ to $(\varepsilon n)^{-1/d'}$. Yet, this process introduces the challenge of protecting the privacy of mean vectors, as detailed in the third step in Algorithms 1 and 3.

Private covariance estimation. Private covariance estimation [18, 45] is closely linked to the private covariance matrix and the private linear projection components of our Algorithm 1. Instead of adding i.i.d. noise, [4, 38] improved the dependence on d in the estimation error by sampling top eigenvectors with the exponential mechanism. However, it requires d' as an input parameter (in our approach, it can be chosen privately) and a lower bound on $\sigma_{d'}(\mathbf{M})$. The dependence on d is a critical aspect in private mean estimation [37, 42], and it is an open question to determine the optimal dependence on d for low-dimensional synthetic data generation.

2. Preliminaries

2.1 *Differential Privacy*

We use the following definition of ε-differential privacy from [20]. Note that in particular, if the algorithm is $\mathscr{A}\colon \Omega^n \to \Omega^m$, then its output is also a dataset of size m, which is generated by $\mathscr{A}$ from the input real dataset. We say the synthetic dataset provides ε-differential privacy if the synthetic data algorithm $\mathscr{A}$ is differentially private.

Definition 2 (Differential privacy). A randomized algorithm $\mathscr{A} : \Omega^n \to \mathscr{R}$ provides *ε-differential privacy* if for any input data D, D' that differs on only one element (or D and D' are adjacent datasets) and for any measurable set $S \subseteq \text{range}(\mathscr{A})$, there is

$$\mathbb{P}\{\mathscr{A}(D) \in S\} \leq e^{\varepsilon} \cdot \mathbb{P}\{\mathscr{A}(D') \in S\}.$$

Here the probability is taken from the probability space of the randomness of $\mathscr{A}$.

For multiple differentially private algorithms, differential privacy has a useful property that their sequential composition is also differentially private [20, Theorem 3.16].

Lemma 1 (Theorem 3.16 in [20]). Suppose $\mathscr{A}_i$ is ε_i-differentially private for $i = 1, \ldots, m$, then the sequential composition $x \mapsto (\mathscr{A}_1(x), \ldots, \mathscr{A}_m(x))$ is $\sum_{i=1}^m \varepsilon_i$-differentially private.

Moreover, the following result about *adaptive composition* indicates that algorithms in a sequential composition can use the outputs in the previous steps:

Lemma 2 (Theorem 1 in [21]). Suppose a randomized algorithm $\mathscr{A}_1(x) : \Omega^n \to \mathscr{R}_1$ is ε_1-differentially private, and $\mathscr{A}_2(x, y) : \Omega^n \times \mathscr{R}_1 \to \mathscr{R}_2$ is ε_2-differentially private with respect to the first component for any fixed y. Then the sequential composition

$$x \mapsto (\mathscr{A}_1(x), \mathscr{A}_2(x, \mathscr{A}_1(x)))$$

is $(\varepsilon_1 + \varepsilon_2)$-differentially private.

Since our method involves private counts of data points, we will use integer Laplacian noise to ensure they are integers.

Definition 3 (Integer Laplacian distribution, [33]). An *integer (or discrete) Laplacian distribution* with parameter σ is a discrete distribution on $\mathbb{Z}$ with probability density function

$$f(z) = \frac{1 - p_\sigma}{1 + p_\sigma} \exp\left(-|z|/\sigma\right), \quad z \in \mathbb{Z},$$

where $p_\sigma = \exp(-1/\sigma)$. A random variable $Z \sim \text{Lap}_{\mathbb{Z}}(\sigma)$ is mean-zero and sub-exponential with variance $\text{Var(Z)} \leq 2\sigma^2$.

2.2 *Wasserstein distance*

The formal definition of p-Wasserstein distance is given as follows:

DEFINITION 4 (p-Wasserstein distance). Consider a metric space (Ω, ρ). The *p-Wasserstein distance* (see e.g. [54] for more details) between two probability measures μ, ν is defined as

$$W_p(\mu,\nu) := \left(\inf_{\gamma\in\Gamma(\mu,\nu)}\int_{\Omega\times\Omega}\rho(x,y)^p \mathrm{d}\gamma(x,y)\right)^{1/p},$$

where $\Gamma(\mu,\nu)$ is the set of all couplings of μ and ν.

In particular, when $p = 1$, the W_1 distance is also known as the earth mover's distance because it is equivalent to the optimal transportation problem if the probability measures are discrete. Furthermore, W_1 has the following Kantorovich-Rubinstein duality (see, e.g. [54]), which gives an equivalent representation with the Lipschitz functions:

$$W_1(\mu,\nu) = \sup_{\mathrm{Lip}(f)\le 1}\left(\int f\mathrm{d}\mu - \int f\mathrm{d}\nu\right).$$

Here the supremum is taken over the set of all 1-Lipschitz functions on Ω.

3. Private linear projection

3.1 *Private centered covariance matrix*

Algorithm 2 Private Covariance Matrix

Input: Matrix $\mathbf{X} = [X_1, \ldots, X_n]$, privacy parameter ε, and variance parameter $\sigma = \frac{3d^2}{\varepsilon n}$.
(Computing the covariance matrix) Compute the mean $\overline{X} = \frac{1}{n}\sum_{i=1}^n X_i$ and the centered covariance matrix $\mathbf{M}$.
(Generating a Laplacian random matrix) Generate i.i.d. independent random variables $\lambda_{ij} \sim \mathrm{Lap}(\sigma), i \le j$. Define a symmetric matrix $\mathbf{A}$ such that

$$\mathbf{A}_{ij} = \mathbf{A}_{ji} = \begin{cases} \lambda_{ij} & \text{if } i < j; \\ 2\lambda_{ii} & \text{if } i = j, \end{cases}$$

Output: The noisy covariance matrix $\widehat{\mathbf{M}} = \mathbf{M} + \mathbf{A}$.

We start with the first step: finding a d'-dimensional private linear affine subspace and projecting $\mathbf{X}$ onto it. Consider the $d \times n$ data matrix $\mathbf{X} = [X_1, \ldots, X_n]$, where $X_1, \ldots, X_n \in \mathbb{R}^d$. The rank of the covariance matrix $\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$ measures the dimension of the *linear subspace* spanned by $X_1, \ldots, X_n$. If we subtract the mean vector and consider the centered covariance matrix $\mathbf{M}$ in (1.2), then the rank of $\mathbf{M}$ indicates the dimension of the *affine linear subspace* that $\mathbf{X}$ lives in.

To guarantee the privacy of $\mathbf{M}$, we add a symmetric Laplacian random matrix $\mathbf{A}$ to $\mathbf{M}$ to create a private Hermitian matrix $\widehat{\mathbf{M}}$ from Algorithm 2. The variance of entries in $\mathbf{A}$ is chosen such that the following privacy guarantee holds:

PROPOSITION 2. Algorithm 2 is ε-differentially private.

Proof. Before applying the definition of differential privacy, we compute the entries of $\mathbf{M}$ explicitly. One can easily check that

$$\mathbf{M} = \frac{1}{n}\sum_{k=1}^{n} X_k X_k^{\mathsf{T}} - \frac{1}{n(n-1)}\sum_{k\neq\ell} X_k X_\ell^{\mathsf{T}}. \tag{3.1}$$

Now, if there are neighboring datasets $\mathbf{X}$ and $\mathbf{X}'$, suppose $X_k = (X_k^{(1)}, \ldots, X_k^{(d)})^{\mathsf{T}}$ is a column vector in $\mathbf{X}$ and $X_k' = (X_k'^{(1)}, \ldots, X_k'^{(d)})^{\mathsf{T}}$ is a column vector in $\mathbf{X}'$, and all other column vectors are the same. Let $\mathbf{M}$ and $\mathbf{M}'$ be the covariance matrix of $\mathbf{X}$ and $\mathbf{X}'$, respectively. Then we consider the density function ratio for the output of Algorithm 2 with input $\mathbf{X}$ and $\mathbf{X}'$:

$$\begin{aligned}
\frac{\mathrm{den}_A(\widehat{\mathbf{M}}-\mathbf{M})}{\mathrm{den}_A(\widehat{\mathbf{M}}-\mathbf{M}')} &= \prod_{i<j}\frac{\mathrm{den}_{\lambda_{ij}}((\widehat{\mathbf{M}}-\mathbf{M})_{ij})}{\mathrm{den}_{\lambda_{ij}}((\widehat{\mathbf{M}}-\mathbf{M}')_{ij})}\prod_{i=j}\frac{\mathrm{den}_{2\lambda_{ij}}((\widehat{\mathbf{M}}-\mathbf{M})_{ij})}{\mathrm{den}_{2\lambda_{ij}}((\widehat{\mathbf{M}}-\mathbf{M}')_{ij})} \\
&= \prod_{i<j}\frac{\exp\left(-\frac{|(\widehat{\mathbf{M}}-\mathbf{M})_{ij}|}{\sigma}\right)}{\exp\left(-\frac{|(\widehat{\mathbf{M}}-\mathbf{M}')_{ij}|}{\sigma}\right)}\prod_{i}\frac{\exp\left(-\frac{|(\widehat{\mathbf{M}}-\mathbf{M})_{ii}|}{2\sigma}\right)}{\exp\left(-\frac{|(\widehat{\mathbf{M}}-\mathbf{M}')_{ii}|}{2\sigma}\right)} \\
&\leq \exp\left(\sum_{i<j}|\mathbf{M}|_{ij}-\mathbf{M}'_{ij}/\sigma + \sum_{i}|\mathbf{M}_{ii}-\mathbf{M}'_{ii}|/(2\sigma)\right) = \exp\left(\frac{1}{2\sigma}\sum_{i,j}|\mathbf{M}|_{ij}-\mathbf{M}'_{ij}\right).
\end{aligned}$$

As the datasets differs on only one data X_k, consider all entry containing X_k in (3.1), we have

$$\begin{aligned}
&|\mathbf{M}_{ij}-\mathbf{M}'_{ij}| \\
&\leq \frac{1}{n}\left|X_k^{(i)}X_k^{(j)} - X_k'^{(i)}X_k'^{(j)}\right| + \frac{1}{n(n-1)}\sum_{\ell\neq k}\left|X_k^{(i)} - X_k'^{(i)}\right|X_\ell^{(j)} + \frac{1}{n(n-1)}\sum_{\ell\neq k}X_\ell^{(i)}\left|X_k^{(j)} - X_k'^{(j)}\right| \\
&\leq \frac{2}{n} + \frac{2}{n(n-1)}\cdot 2(n-1) = \frac{6}{n}.
\end{aligned}$$

Therefore, substituting the result in the probability ratio implies

$$\frac{\mathrm{den}_A(\widehat{\mathbf{M}}-\mathbf{M})}{\mathrm{den}_A(\widehat{\mathbf{M}}-\mathbf{M}')} \leq \exp\left(\frac{1}{2\sigma}\cdot d^2\cdot\frac{6}{n}\right) = \exp\left(\frac{3d^2}{\sigma n}\right),$$

and when $\sigma = \frac{3d^2}{\varepsilon n}$, Algorithm 2 is ε-differentially private. □

3.2 *Noisy projection*

The private covariance matrix $\widehat{\mathbf{M}}$ induces private subspaces spanned by eigenvectors of $\widehat{\mathbf{M}}$. We then perform a truncated SVD on $\widehat{\mathbf{M}}$ to find a private d'-dimensional subspace $\widehat{\mathbf{V}}_{d'}$ and project original data

onto $\widehat{\mathbf{V}}_{d'}$. Here, the matrix $\widehat{\mathbf{V}}_{d'}$ also indicates the subspace generated by its orthonormal columns. The full steps are summarized in Algorithm 3.

Algorithm 3 Noisy Projection

Input: True data matrix $\mathbf{X} = [X_1, \ldots, X_n]$, $X_i \in [0,1]^d$, privacy parameters ε, the private covariance matrix $\widehat{\mathbf{M}}$ from Algorithm 2, and a target dimension d'.
(Singular value decomposition) Compute the top d' orthonormal eigenvectors $\widehat{v}_1, \ldots, \widehat{v}_{d'}$ of $\widehat{\mathbf{M}}$ and denote $\widehat{\mathbf{V}}_{d'} = [\widehat{v}_1, \ldots, \widehat{v}_{d'}]$.
(Private centering) Compute $\overline{X} = \frac{1}{n}\sum_{i=1}^n X_i$. Let $\lambda \in \mathbb{R}^d$ be a random vector with i.i.d. components of $\mathrm{Lap}(d/(\varepsilon n))$. Shift each X_i to $X_i - (\overline{X} + \lambda)$ for $i \in [n]$.
(Projection) Project $\{X_i - (\overline{X} + \lambda)\}_{i=1}^n$ onto the linear subspace spanned by $\widehat{v}_1, \ldots, \widehat{v}_{d'}$. The projected data $\widehat{X}_i$ is given by $\widehat{X}_i = \sum_{j=1}^{d'} \left\langle X_i - (\overline{X} + \lambda), \widehat{v}_j \right\rangle \widehat{v}_j$.
Output: The data matrix after projection $\widehat{\mathbf{X}} = [\widehat{X}_1 \ldots \widehat{X}_n]$.

Algorithm 3 only guarantees private basis $\widehat{v}_1, \ldots, \widehat{v}_{d'}$ for each $\widehat{X}_i$, but the coordinates of $\widehat{X}_i$ in terms of $\widehat{v}_1, \ldots, \widehat{v}_{d'}$ are *not private*. Algorithms 4 and 5 in the next stage will output synthetic data on the private subspace $\widehat{\mathbf{V}}_{d'}$ based on $\widehat{\mathbf{X}}$. The privacy analysis combines the two stages based on Lemma 2, and we state the results in Section 4.

3.3 *Accuracy guarantee for noisy projection*

The data matrix $\widehat{\mathbf{X}}$ corresponds to an empirical measure $\mu_{\widehat{\mathbf{X}}}$ supported on the subspace $\widehat{\mathbf{V}}_d$. In this subsection, we characterize the 1-Wasserstein distance between the empirical measure $\mu_{\widehat{\mathbf{X}}}$ and the empirical measure of the centered dataset $\mathbf{X} - \overline{X}\mathbf{1}^\mathsf{T}$, where $\mathbf{1} \in \mathbb{R}^n$ is the all-1 vector. This problem can be formulated as the stability of a low-rank projection based on a covariance matrix with additive noise. We first provide the following useful deterministic lemma.

LEMMA 3 (Stability of noisy projection). Let $\mathbf{X}$ be a $d \times n$ matrix and $\mathbf{A}$ be a $d \times d$ Hermitian matrix. Let $\mathbf{M} = \frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$ with eigenvalues $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d$. Let $\widehat{\mathbf{M}} = \frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T} + \mathbf{A}$, $\widehat{\mathbf{V}}_{d'}$ be a $d \times d'$ matrix whose columns are the first d' orthonormal eigenvectors of $\widehat{\mathbf{M}}$, and $\mathbf{Y} = \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}\mathbf{X}$. Let $\mu_{\mathbf{X}}$ and $\mu_{\mathbf{Y}}$ be the empirical measures of column vectors of $\mathbf{X}$ and $\mathbf{Y}$, respectively. Then

$$W_2^2(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \leq \frac{1}{n}\|\mathbf{X} - \mathbf{Y}\|_F^2 \leq \sum_{i>d'} \sigma_i + 2d'\|\mathbf{A}\|. \tag{3.2}$$

Proof. Let $\widehat{v}_1, \ldots, \widehat{v}_d$ be a set of orthonormal eigenvectors for $\widehat{\mathbf{M}}$ with the corresponding eigenvalues $\widehat{\sigma}_1, \ldots, \widehat{\sigma}_d$. Define four matrices whose column vectors are eigenvectors:

$$\mathbf{V} = [v_1, \ldots, v_d], \qquad \widehat{\mathbf{V}} = [\widehat{v}_1, \ldots, \widehat{v}_d],$$
$$\mathbf{V}_{d'} = [v_1, \ldots, v_{d'}], \qquad \widehat{\mathbf{V}}_{d'} = [\widehat{v}_1, \ldots, \widehat{v}_{d'}].$$

By orthogonality, the following identities hold:

$$\sum_{i=1}^{d} \|v_i^\mathsf{T}\mathbf{X}\|_2^2 = \sum_{i=1}^{d} \|\widehat{v}_i^\mathsf{T}\mathbf{X}\|_2^2 = \|\mathbf{X}\|_F^2,$$
$$\sum_{i>d'} \|v_i^\mathsf{T}\mathbf{X}\|_2^2 = \|\mathbf{X} - \mathbf{V}_{d'}\mathbf{V}_{d'}^\mathsf{T}\mathbf{X}\|_F^2,$$
$$\sum_{i>d'} \|\widehat{v}_i^\mathsf{T}\mathbf{X}\|_2^2 = \|\mathbf{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}\mathbf{X}\|_F^2.$$

Separating the top d' eigenvectors from the rest, we obtain

$$\sum_{i\le d'} \|v_i^\mathsf{T}\mathbf{X}\|_2^2 + \|\mathbf{X} - \mathbf{V}_{d'}\mathbf{V}_{d'}^\mathsf{T}\mathbf{X}\|_F^2 = \sum_{i\le d'} \|\widehat{v}_i^\mathsf{T}\mathbf{X}\|_2^2 + \|\mathbf{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}\mathbf{X}\|_F^2.$$

Therefore

$$\begin{aligned}
\|\mathbf{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}\mathbf{X}\|_F^2 &= \sum_{i\le d'} \|v_i^\mathsf{T}\mathbf{X}\|_2^2 - \sum_{i\le d'} \|\widehat{v}_i^\mathsf{T}\mathbf{X}\|_2^2 + \|\mathbf{X} - \mathbf{V}_{d'}\mathbf{V}_{d'}^\mathsf{T}\mathbf{X}\|_F^2 \\
&= n\sum_{i\le d'} \sigma_i - n\sum_{i\le d'} \widehat{v}_i^\mathsf{T}\mathbf{M}\widehat{v}_i + n\sum_{i>d'} \sigma_i \\
&= n\sum_{i\le d'} \sigma_i - n\sum_{i\le d'} \widehat{v}_i^\mathsf{T}(\widehat{\mathbf{M}} - \mathbf{A})\widehat{v}_i + n\sum_{i>d'} \sigma_i \\
&= n\sum_{i\le d'} (\sigma_i - \widehat{\sigma}_i) + n\,\mathrm{tr}(\mathbf{A}\widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}) + n\sum_{i>d'} \sigma_i. \qquad (3.3)
\end{aligned}$$

By Weyl's inequality, for $i \le d'$,

$$|\sigma_i - \widehat{\sigma}_i| \le \|\mathbf{A}\|. \qquad (3.4)$$

By von Neumann's trace inequality,

$$\mathrm{tr}(A\widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}) \le \sum_{i=1}^{d'} \sigma_i(\mathbf{A}). \qquad (3.5)$$

From (3.3), (3.4) and (3.5),

$$\frac{1}{n}\|\mathbf{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}\mathbf{X}\|_F^2 \le \sum_{i>d'} \sigma_i + d'\|\mathbf{A}\| + \sum_{i=1}^{d'} \sigma_i(\mathbf{A}) \le \sum_{i>d'} \sigma_i + 2d'\|\mathbf{A}\|.$$

Let Y_i be the i-th column of $\mathbf{Y}$. We have

$$W_2^2(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \le \frac{1}{n}\sum_{i=1}^{n} \|X_i - Y_i\|_2^2 = \frac{1}{n}\|\mathbf{X} - \mathbf{Y}\|_F^2.$$

Therefore, (3.2) holds. □

Note that inequality (3.2) holds without any spectral gap assumption on $\mathbf{M}$. Applying Davis-Kahan inequality would require $\sigma_{d'} - \sigma_{d'+1}$ to be large while Lemma 3 is applicable even when $\sigma_{d'} = \sigma_{d'+1}$. In the context of sample covariance matrices for random datasets, a related bound without a spectral gap condition is derived in [48, Proposition 2.2]. Furthermore, Lemma 3 bears a conceptual resemblance to [3, Theorem 5], which deals with low-rank matrix approximation under perturbation. With Lemma 3, we derive the following Wasserstein distance bounds between the centered dataset $\mathbf{X} - \overline{X}\mathbf{1}^\mathsf{T}$ and the dataset $\widehat{\mathbf{X}}$.

PROPOSITION 3. For input data $\mathbf{X}$ and output data $\widehat{\mathbf{X}}$ in Algorithm 3, let $\mathbf{M}$ be the covariance matrix defined in (1.2). Assume $n \ge 1/\varepsilon$. Then for an absolute constant $C > 0$,

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) \le \left(\mathbb{E}\, W_2^2(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}})\right)^{1/2} \le \sqrt{2\sum_{i>d'}\sigma_i(\mathbf{M})} + \sqrt{\frac{Cd'd^{2.5}}{\varepsilon n}}.$$

Proof. For the true covariance matrix $\mathbf{M}$, consider its SVD

$$\mathbf{M} = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})(X_i - \overline{X})^\mathsf{T} = \sum_{j=1}^{d}\sigma_j v_j v_j^\mathsf{T}, \tag{3.6}$$

where $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_d$ are the singular values and $v_1 \ldots v_d$ are corresponding orthonormal eigenvectors. Moreover, define two $d \times d'$ matrices

$$\mathbf{V}_{d'} = [v_1, \ldots, v_{d'}], \qquad \widehat{\mathbf{V}}_{d'} = [\widehat{v}_1, \ldots, \widehat{v}_{d'}].$$

Then the matrix $\widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}$ is a projection onto the subspace spanned by the principal components $\widehat{v}_1, \ldots, \widehat{v}_{d'}$.

In Algorithm 3, for any data X_i we first shift it to $X_i - \overline{X} - \lambda$ and then project it to $\widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}(X_i - \overline{X} - \lambda)$. Therefore,

$$\begin{aligned}\|X_i - \overline{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}(X_i - \overline{X} - \lambda)\|_\infty &\le \|X_i - \overline{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}(X_i - \overline{X})\|_\infty + \|\widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}\lambda\|_\infty \\ &\le \|X_i - \overline{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\mathsf{T}(X_i - \overline{X})\|_2 + = \|\lambda\|_2.\end{aligned}$$

Let Z_i denote $X_i - \overline{X}$ and $\mathbf{Z} = [Z_1, \ldots, Z_n]$. Then

$$\frac{1}{n}\mathbf{Z}\mathbf{Z}^\mathsf{T} = \frac{n-1}{n}\mathbf{M}.$$

With Lemma 3, by definition of the Wasserstein distance, we have

$$\begin{aligned} W_2^2(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\intercal}, \mu_{\widehat{\mathbf{X}}}) &\le \frac{1}{n}\sum_{i=1}^{n} \|X_i - \overline{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\intercal(X_i - \overline{X} - \lambda)\|_\infty^2 \\ &\le \frac{2}{n}\sum_{i=1}^{n} \|X_i - \overline{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\intercal(X_i - \overline{X})\|_2^2 + 2\|\lambda\|_2^2 \\ &= \frac{2}{n}\|\mathbf{Z} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^\intercal\mathbf{Z}\|_F^2 + 2\|\lambda\|_2^2 \\ &\le 2\sum_{i=d'}^{n}\sigma_i(\mathbf{M}) + 4d'\|\mathbf{A}\| + 2\|\lambda\|_2^2. \end{aligned} \tag{3.7}$$

Since $\lambda = (\lambda_1, \ldots, \lambda_d)$ is a Laplacian random vector with i.i.d. Lap$(1/(\varepsilon n))$ entries,

$$\mathbb{E}\,\|\lambda\|_2^2 = \sum_{j=1}^{d}\mathbb{E}\,|\lambda_j|^2 = \frac{2d}{\varepsilon^2 n^2}. \tag{3.8}$$

Furthermore, in Algorithm 2, A is a symmetric random matrix with independent Laplacian random variables on and above its diagonal. Thus, we have the tail bound for its norm [16, Theorem 1.1]

$$\mathbb{P}\left\{\|\mathbf{A}\| \ge \sigma(C\sqrt{d} + t)\right\} \le C_0\exp(-C_1\min(t^2/4, t/2)). \tag{3.9}$$

And we can further compute the expectation bound for $\|\mathbf{A}\|$ from (3.9) with the choice of $\sigma = \frac{3d^2}{\varepsilon n}$,

$$\mathbb{E}\,\|\mathbf{A}\| \le C\sigma\sqrt{d} + \int_0^\infty C_0\exp\left(-C_1\min\left(\frac{t^2}{4\sigma^2}, \frac{t}{2\sigma}\right)\right)\mathrm{d}t \lesssim \frac{d^{2.5}}{\varepsilon n}.$$

Combining the two bounds above and (3.7), as the 1-Wasserstein distance is bounded by the 2-Wasserstein distance and inequality $\sqrt{x+y} \le \sqrt{x} + \sqrt{y}$ holds for all $x, y \ge 0$,

$$\begin{aligned} \mathbb{E}\,W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\intercal}, \mu_{\widehat{\mathbf{X}}}) &\le \left(\mathbb{E}\,W_2^2(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\intercal}, \mu_{\widehat{\mathbf{X}}})\right)^{1/2} \\ &\le \sqrt{2\sum_{i>d'}\sigma_i(\mathbf{M})} + \sqrt{4d'\,\mathbb{E}\,\|\mathbf{A}\|} + \sqrt{2\,\mathbb{E}\,\|\lambda\|_2^2} \\ &\le \sqrt{2\sum_{i>d'}\sigma_i(\mathbf{M})} + \sqrt{\frac{Cd'd^{2.5}}{\varepsilon n}}, \end{aligned}$$

where the last inequality holds under the assumption $\varepsilon n \ge 1$. □

4. Synthetic data subroutines

In the next stage of Algorithm 1, we construct synthetic data on the private subspace $\widehat{\mathbf{V}}_{d'}$. Since the original data X_i is in $[0,1]^d$, after Algorithm 3, we have

$$\|\widehat{X}_i\|_2 = \|X_i - \overline{X} - \lambda\|_2 \le \sqrt{d} + \|\overline{X} + \lambda\|_2 =: R$$

for any fixed $\lambda \in \mathbb{R}^d$. Therefore, the data after projection would lie in a d'-dimensional ball embedded in $\mathbb{R}^d$ with radius R, and the domain for the subroutine is

$$\Omega' = \{a_1\widehat{v}_1 + \cdots + a_{d'}\widehat{v}_{d'} \mid a_1^2 + \cdots + a_{d'}^2 \le R^2\},$$

where $\widehat{v}_1, \ldots, \widehat{v}_{d'}$ are the first d' private principal components in Algorithm 3. Depending on whether $d' = 2$ or $d' \ge 3$, we apply two different algorithms from [31]: private measure mechanism (PMM) and private signed measure mechanism (PSMM).

A major difference between the two methods is the partition step. PMM uses a hierarchical binary partition of the entire space into r layers, while PSMM partitions the entire space into disjoint regions. When $d' = 2$, PMM has a better accuracy rate while when $d' \ge 3$, PSMM has a better dependence on d' in the accuracy bound; See Remark 1 for more details.

4.1 $d' = 2$: *PMM*

The synthetic data subroutine Algorithm 4 is adapted from the PMM in [31, Algorithm 4]. The PMM algorithm generates synthetic data in a hypercube by first partition the cube and then perturb the count in each sub-regions. It involves a certain partition structure, *binary hierarchical partition*.

Definition 5 (Binary hierarchical partition, [31]). A *binary hierarchical partition* of a set Ω of depth r is a family of subsets Ω_θ indexed by $\theta \in \{0,1\}^{\le r}$, where

$$\{0,1\}^{\le k} = \{0,1\}^0 \sqcup \{0,1\}^1 \sqcup \cdots \sqcup \{0,1\}^k, \quad k = 0,1,2\ldots,$$

and such that Ω_θ is partitioned into $\Omega_{\theta 0}$ and $\Omega_{\theta 1}$ for every $\theta \in \{0,1\}^{\le r-1}$. By convention, the cube $\{0,1\}^0$ corresponds to $\emptyset$ and we write $\Omega_\emptyset = \Omega$.

The detailed description of Algorithm 4 is as follows. The privacy and accuracy guarantees of Algorithm 4 are proved in the next proposition after stating the algorithm.

For the new region Ω' where projected data located, we first enlarge this ℓ_2-ball of radius R into a hypercube Ω_{PMM} of edge length $2R$ defined in Algorithm 4. Both the ℓ_2-ball Ω' and the larger hypercube Ω_{PMM} are inside the subspace $\widehat{\mathbf{V}}_{d'}$.

Next, for the hypercube Ω_{PMM}, we are going to run PMM in [31]. We obtain a binary hierarchical partition $\{\Omega_\theta\}_{\theta\in\{0,1\}^{\le r}}$ for $r = \lceil \log_2(\varepsilon n)\rceil$ by doing equal divisions of the hypercube recursively for r rounds. Each round after the division, we count the data points in every new subregion Ω_θ and add integer Laplacian noise to it.

Finally, a consistency step ensures the output is a well-defined probability measure. Here, the counts are considered to be *consistent* if they are non-negative and the counts of two smaller subregions $\Omega_{\theta 0}, \Omega_{\theta 1}$ can add up to the counts of the larger regions Ω_θ containing them. We refer the readers to [31] for more detailed procedures of this step.

Algorithm 4 PMM Subroutine

Input: dataset $\widehat{\mathbf{X}} = (\widehat{X}_1, \ldots, \widehat{X}_n)$ in the region

$$\Omega' = \{a_1\widehat{v}_1 + \cdots + a_{d'}\widehat{v}_{d'} \mid a_1^2 + \cdots + a_{d'}^2 \le R\}.$$

(Binary partition) Let $r = \lceil \log_2(\varepsilon n) \rceil$ and $\sigma_j = \varepsilon^{-1} \cdot 2^{\frac{1}{2}\left(1-\frac{1}{d'}\right)(r-j)}$. Enlarge the region Ω' into

$$\Omega_{\mathrm{PMM}} = \{a_1\widehat{v}_1 + \cdots + a_{d'}\widehat{v}_{d'} \mid a_i \in [-R, R], \forall i \in [d']\}.$$

Build a binary partition $\{\Omega_\theta\}_{\theta\in\{0,1\}^{\le r}}$ on Ω_{PMM}.
(Noisy count) For any θ, count the number of data in the region Ω_θ denoted by $n_\theta = \left|\widehat{\mathbf{X}} \cap \Omega_\theta\right|$, and let $n'_\theta = (n_\theta + \lambda_\theta)_+$, where λ_θ are independent integer Laplacian random variables with $\lambda \sim \mathrm{Lap}_{\mathbb{Z}}(\sigma_{|\theta|})$, and $|\theta|$ is the length of the vector θ.
(Consistency) Enforce consistency of $\{n'_\theta\}_{\theta\in\{0,1\}^{\le r}}$.
Output: Synthetic data $\mathbf{X}'$ generated by selecting n'_θ many data points arbitrarily (independently of $\widehat{\mathbf{X}}$) from Ω_θ for every $\theta \in \{0,1\}^r$.

Proposition 4. The subroutine Algorithm 4 is ε-differentially private. Assume $n \ge 1/\varepsilon$. For any $d' \ge 2$, with the input as the projected data $\widehat{\mathbf{X}}$ and the range Ω' with radius R, Algorithm 4 has an accuracy bound

$$\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) \le CR(\varepsilon n)^{-1/d'},$$

where the expectation is taken with respect to the randomness of the synthetic data subroutine, conditioned on R.

Proof. The privacy guarantee follows from [31, Theorem 1.1]. For accuracy, note that the region Ω' is a subregion of a d'-dimensional ball. Algorithm 4 enlarges the region to a d'-dimensional hypercube with side length $2R$. By re-scaling the size of the hypercube and applying [31, Corollary 4.4], we obtain the accuracy bound. □

4.2 $d' \ge 3$: *PSMM*

The PSMM introduced in [31] generates a synthetic dataset $\mathbf{Y}$ in a compact domain Ω whose empirical measure $\mu_{\mathbf{Y}}$ is close to the empirical measure $\mu_{\mathbf{X}}$ of the original dataset $\mathbf{X}$ under the 1-Wasserstein distance.

PSMM runs in polynomial time, and the main steps are as follows. We first partition the domain Ω into m disjoint subregions $\Omega_1, \ldots, \Omega_m$ and count the number of data points in each subregion. Then, we perturb the counts in each subregion with i.i.d. integer Laplacian noise. Based on the noisy counts, one can approximate $\mu_{\mathbf{X}}$ with a signed measure ν supported on m points. Then, we find the closest probability measure $\hat{\nu}$ to the signed measure ν under the bounded Lipschitz distance by solving a linear programming problem.

We provide the main steps of PSMM in Algorithm 5. Details about the linear programming in the *synthetic probability measure* step can be found in [31]. We apply PSMM from [31] when the metric space is an ℓ_2-ball of radius R inside $\widehat{\mathbf{V}}_{d'}$ and the following privacy and accuracy guarantees hold:

Algorithm 5 PSMM Subroutine

Input: dataset $\widehat{\mathbf{X}} = (\widehat{X}_1, \ldots, \widehat{X}_n)$ in the region

$$\Omega' = \{a_1\widehat{v}_1 + \cdots + a_{d'}\widehat{v}_{d'} \mid a_1^2 + \cdots + a_{d'}^2 \le R^2\}.$$

(Integer lattice) Let $\delta = \sqrt{d/d'}(\varepsilon n)^{-1/d'}$. Find the lattice over the region:

$$L = \{a_1\widehat{v}_1 + \cdots + a_{d'}\widehat{v}_{d'} \mid a_1^2 + \ldots 7 + a_{d'}^2 \le R^2, a_1, \ldots, a_{d'} \in \delta\mathbb{Z}\}.$$

(Counting) For any $v = a_1\widehat{v}_1 + \cdots + a_{d'}\widehat{v}_{d'} \in L$, count the number

$$n_v = \left|\widehat{\mathbf{X}} \cap \{b_1\widehat{v}_1 + \cdots + b_{d'}\widehat{v}_{d'} \mid b_i \in [a_i, a_i + \delta)\}\right|.$$

(Adding noise) Define a synthetic signed measure ν such that for any $v \in L$,

$$\nu(\{v\}) = (n_v + \lambda_v)/n,$$

where $\lambda_v \sim \mathrm{Lap}_{\mathbb{Z}}(1/\varepsilon)$, $v \in L$ are i.i.d. random variables.
(Synthetic probability measure) Use linear programming and find the closest probability measure $\widehat{\nu}$ to ν under the bounded Lipschitz distance.
Output: Synthetic data $\mathbf{X}'$ containing copies of elements in L so that $\mu_{\mathbf{X}'}$ and $\widehat{\nu}$ are arbitrarily close (such $\mathbf{X}'$ exist when the size of $\mathbf{X}'$ is large enough; see (31, Section 3)).

Proposition 5. The subroutine Algorithm 5 is ε-differentially private. Assume $n \ge 1/\varepsilon$. When $d' \ge 3$, with the input as the projected data $\widehat{\mathbf{X}}$ and the range Ω' with radius R, the algorithm has an accuracy bound

$$\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) \lesssim \frac{R}{\sqrt{d'}}(\varepsilon n)^{-1/d'}, \tag{4.1}$$

where the expectation is conditioned on R.

Proof. The proposition is a direct corollary to the result in [31]. The size of the scaled integer lattice $\delta\mathbb{Z}$ in the unit d-dimensional ball of radius R is bounded by $(\frac{C}{\delta R})^d$ for an absolute constant $C > 0$ (see, e.g. [26, Claim 2.9] and [11, Proposition 3.7]). Then, the number of subregions in Algorithm 5 is bounded by

$$|L| \le \left(\frac{R}{\sqrt{d'}} \cdot \frac{C}{\delta}\right)^{d'}.$$

By [31, Theorem 3.6], we have

$$\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) \le \delta + \frac{2}{\varepsilon n}\left(\frac{R}{\sqrt{d'}} \cdot \frac{C}{\delta}\right)^{d'} \cdot \frac{1}{d'}\left(\left(\frac{R}{\sqrt{d'}} \cdot \frac{C}{\delta}\right)^{d'}\right)^{-\frac{1}{d'}}.$$

Taking $\delta = \frac{CR}{\sqrt{d'}}(\varepsilon n)^{-\frac{1}{d'}}$ concludes the proof. □

REMARK 1 (PMM vs PSMM for $d' \geq 2$). For general $d' \geq 2$, PMM can still be applied, and the accuracy bound becomes $\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) \leq CR(\varepsilon n)^{-1/d'}$. Compared to (4.1), a the accuracy bound from PMM is weaker by a factor of $\sqrt{d'}$. However, as shown in [31], PMM has a running time linear in n and d, which is more computationally efficient than PSMM given in Algorithm 5 with running time polynomial in n, d.

4.3 *Adding a private mean vector and metric projection*

After generating the private synthetic data, since we shift the data by its private mean before projection, we need to add another private mean vector back, which shifts the dataset $\widehat{\mathbf{X}}$ to a new private affine subspace close to the original dataset $\mathbf{X}$. The output data vectors in $\mathbf{X}''$ (defined in Algorithm 1) are not necessarily inside $[0,1]^d$. The subsequent metric projection enforces all synthetic data inside $[0,1]^d$. Importantly, this post-processing step does not have privacy costs.

After metric projection, dataset $\mathbf{Y}$ from the output of Algorithm 1 is close to an affine subspace, as shown in the next proposition. Notably, (4.2) shows that the metric projection step does not cause the largest accuracy loss among all subroutines.

PROPOSITION 6 ($\mathbf{Y}$ is close to an affine subspace). The function $f : \mathbb{R}^d \to [0,1]^d$ in Algorithm 1 is the metric projection to $[0,1]^d$ with respect to $\|\cdot\|_\infty$, and the accuracy error for the metric projection step in Algorithm 1 is dominated by the error of the previous steps:

$$W_1(\mu_{\mathbf{Y}}, \mu_{\mathbf{X}''}) \leq W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}), \tag{4.2}$$

where the dataset $\mathbf{X}''$ defined in Algorithm 1 is in a d'-dimensional affine subspace.

Proof. For the function f defined in Algorithm 1, we know $f(x)$ is the closest real number to x in the region $[0,1]$ for any $x \in \mathbb{R}$. Furthermore, if $v \in \mathbb{R}^d$ is a vector, then $f(v)$ is the closest vector to v in $[0,1]^d$ with respect to $\|\cdot\|_\infty$. Thus $f : \mathbb{R}^d \to [0,1]^d$ is indeed a metric projection to $[0,1]^d$.

We first assume that the synthetic data $\mathbf{X}''$ also has size n. Then for any column vector X_i'', we know that $Y_i = f(X_i'')$ is its closest vector in $[0,1]^d$ under the ℓ^∞ metric. For the data $X_1, X_2, \ldots, X_n$, suppose that the solution to the optimal transportation problem for $W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''})$ is to match $X_{\tau(i)}$ with X_i'', where τ is a permutation on $[n]$. Then

$$W_1(\mu_{\mathbf{Y}}, \mu_{\mathbf{X}''}) \leq \frac{1}{n}\sum_{i=1}^{n} \|Y_i - X_i''\|_\infty \leq \frac{1}{n}\sum_{i=1}^{n} \|X_{\tau(i)} - X_i''\|_\infty = W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}).$$

In general, if the synthetic dataset has m data points and $m \neq n$, we can split the points and regard both the true dataset and synthetic dataset as of size mn, then it's easy to check that the inequality still holds. □

5. Privacy and accuracy of Algorithm 1

In this section, we summarize the privacy and accuracy guarantees of Algorithm 1. The privacy guarantee is proved by analyzing three parts of our algorithms: private mean, private linear subspace and private data on an affine subspace.

PROPOSITION 7 (Privacy). Algorithm 1 is ε-differentially private.

Proof. We can decompose Algorithm 1 into the following steps:

1. $\mathscr{A}_1(\mathbf{X}) = \widehat{\mathbf{M}}$ is to compute the private covariance matrix with Algorithm 2.
2. $\mathscr{A}_2(\mathbf{X}) = \overline{X} + \lambda$ is to compute the private sample mean.
3. $\mathscr{A}_3(\mathbf{X}, y, \Sigma)$ for fixed y and Σ, is to project the shifted data $\{X_i - y\}_{i=1}^n$ to the first d' principal components of Σ and apply a certain differentially private subroutine (we choose y and Σ as the output of $\mathscr{A}_2$ and $\mathscr{A}_1$, respectively). This step outputs synthetic data $\mathbf{X}' = (X_1', \ldots, X_m')$ on a linear subspace.
4. $\mathscr{A}_4(\mathbf{X}, \mathbf{X}')$ is to shift the dataset to $\{X_i' + \overline{X}_{\text{priv}}\}_{i=1}^m$, where $\overline{X}_{\text{priv}}$ is the private mean vector of the true data step computed by $\mathscr{A}_2$.
5. Metric projection.

It suffices to show that the data before metric projection has already been differentially private. We will need to apply Lemma 2 several times.

With respect to the input $\mathbf{X}$ while fixing other input variables, we know that $\mathscr{A}_1, \mathscr{A}_2, \mathscr{A}_3, \mathscr{A}_4$ are all $\varepsilon/4$-differentially private. Therefore, by using Lemma 2 iteratively, the composition algorithm

$$\mathscr{A}_4(\mathbf{X}, \mathscr{A}_3(\mathbf{X}, \mathscr{A}_2(\mathbf{X}), \mathscr{A}_1(\mathbf{X})))$$

satisfies ε-differential privacy. Hence Algorithm 1 is ε-differentially private. □

The next theorem combines errors from linear projection, synthetic data subroutine using PMM or PSMM, and the post-processing error from mean shift and metric projection.

PROPOSITION 8 (Accuracy). For any given $2 \le d' \le d$ and $n \ge 1/\varepsilon$, the output data $\mathbf{Y}$ from Algorithm 1 with the input data $\mathbf{X}$ satisfies

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \lesssim \sqrt{\sum_{i>d'} \sigma_i(\mathbf{M})} + \sqrt{\frac{d' d^{2.5}}{\varepsilon n}} + \sqrt{\frac{d}{d'}} (\varepsilon n)^{-1/d'}, \tag{5.1}$$

where $\mathbf{M}$ denotes the covariance matrix in (1.2).

Proof. In the case of $n < 1/\varepsilon$, we have $W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \le 1 \le (\varepsilon n)^{-1/d'}$. The result is trivial. We assume $n \ge 1/\varepsilon$ in the rest of the proof.

Similar to privacy analysis, we will decompose the algorithm into several steps. Suppose that

1. $\mathbf{X} - (\overline{X} + \lambda)\mathbf{1}^\mathsf{T}$ denotes the shifted data $\{X_i - \overline{X} - \lambda\}_{i=1}^n$;
2. $\widehat{\mathbf{X}}$ is the data after projection to the private linear subspace;
3. $\mathbf{X}'$ is the output of the synthetic data subroutine in Section 4;
4. $\mathbf{X}'' = \mathbf{X}' + (\overline{X} + \lambda')\mathbf{1}^\mathsf{T}$ denotes the data shifted back;
5. $\mathscr{A}(\mathbf{X})$ is the data after metric projection, which is the output of the whole algorithm.

For the metric projection step, by Proposition 6, we have that

$$W_1(\mu_{\mathbf{X}}, \mu_{\mathscr{A}(\mathbf{X})}) \le W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}) + W_1(\mu_{\mathbf{X}''}, \mu_{\mathscr{A}(\mathbf{X})}) \le 2W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}). \tag{5.2}$$

Moreover, applying the triangle inequality of Wasserstein distance to the other steps of the algorithm, we have

$$\begin{aligned} W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}) &= W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\top}, \mu_{\mathbf{X}'+\lambda'\mathbf{1}^\top}) \\ &\le W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\top}, \mu_{\widehat{\mathbf{X}}}) + W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) + W_1(\mu_{\mathbf{X}'}, \mu_{\mathbf{X}'+\lambda'}) \\ &\le W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\top}, \mu_{\widehat{\mathbf{X}}}) + W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) + \|\lambda'\|_\infty. \end{aligned} \tag{5.3}$$

Note that $W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\top}, \mu_{\widehat{\mathbf{X}}})$ is the projection error we bound in Theorem 3 with $n \ge 1/\varepsilon$, and $W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'})$ is treated in the accuracy analysis of subroutines in Section 4. Moreover, we have

$$\begin{aligned} \mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) &= \mathbb{E}_R\, \mathbb{E}_{\mathbf{X}'}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) \\ &\le \mathbb{E}_R\, \frac{CR}{\sqrt{d'}} (\varepsilon n)^{-1/d'} \\ &\le \frac{C(2\sqrt{d} + \mathbb{E}\,\|\lambda\|_2)}{\sqrt{d'}} (\varepsilon n)^{-1/d'} \\ &\lesssim \sqrt{\frac{d}{d'}} (\varepsilon n)^{-1/d'}. \end{aligned}$$

Here in the last step we use $\mathbb{E}\,\|\lambda\|_2 \le \frac{C\sqrt{d}}{\varepsilon n}$ in (3.8). Since λ' is a sub-exponential random vector, the following bound also holds for some absolute constant $C > 0$:

$$\mathbb{E}\,\|\lambda'\|_\infty \le \frac{C \log d}{\varepsilon n}. \tag{5.4}$$

Hence

$$\begin{aligned} \mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathscr{A}(\mathbf{X})}) &\le 2\,\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}'+(\overline{X}+\lambda')\mathbf{1}^\top}) \\ &\le 2\,\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\top}, \mu_{\widehat{\mathbf{X}}}) + 2\,\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) + 2\,\mathbb{E}\,\|\lambda'\|_\infty \\ &\le 2\sqrt{2\sum_{i>d'} \sigma_i(\mathbf{M})} + 2\sqrt{\frac{Cd'd^{2.5}}{\varepsilon n}} + 2C\sqrt{\frac{d}{d'}} (\varepsilon n)^{-1/d'} + \frac{2C\log d}{\varepsilon n} \\ &\lesssim \sqrt{\sum_{i>d'} \sigma_i(\mathbf{M})} + \sqrt{\frac{d}{d'}} (\varepsilon n)^{-1/d'} + \sqrt{\frac{d'd^{2.5}}{\varepsilon n}}, \end{aligned}$$

where the first inequality is from (5.2), the second inequality is from (5.3) and the third inequality is due to Theorem 3, Proposition 4 and Proposition 5. □

6. Adaptive and private choice of d'

In our main Algorithm 1, d' is regarded as a fixed input hyper-parameter. In this section, we will show that it is possible to choose d' privately without sacrificing accuracy.

LEMMA 4. For $\mathbf{M}$ and $\widehat{\mathbf{M}}$ defined in Algorithm 2, with probability at least $1 - C\exp(-c\sqrt{d})$, there is

$$\left|\sum_{i>d'}\sigma_i(\widehat{\mathbf{M}}) - \sum_{i>d'}\sigma_i(\mathbf{M})\right| \lesssim \frac{(d-d')d^{2.5}}{\varepsilon n}.$$

Proof. By Weyl's inequality, $|\sigma_i(\widehat{\mathbf{M}}) - \sigma_i(\mathbf{M})| \leq \|\mathbf{A}\|$. Applying the (3.9) of the noise $\mathbf{A}$ implies the inequality in the lemma. □

Therefore, from Proposition 8, with probability at least $1 - C\exp(-c\sqrt{d})$, we have the following accuracy bound:

$$\begin{aligned} W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) &\lesssim \sqrt{\sum_{i>d'}\sigma_i(\mathbf{M})} + \sqrt{\frac{d'd^{2.5}}{\varepsilon n}} + \sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'} \\ &\lesssim \sqrt{\sum_{i>d'}\sigma_i(\widehat{\mathbf{M}}) + \frac{(d-d')d^{2.5}}{\varepsilon n}} + \sqrt{\frac{d'd^{2.5}}{\varepsilon n}} + \sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'} \\ &\lesssim \sqrt{\sum_{i>d'}\sigma_i(\widehat{\mathbf{M}})} + \sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'} + \sqrt{\frac{d^{3.5}}{\varepsilon n}}. \end{aligned}$$

Since the last term above is not related to d', we can choose

$$d' := \arg\min_{2\leq k\leq d}\left(\sqrt{\sum_{i>k}\sigma_i(\widehat{\mathbf{M}})} + \sqrt{\frac{d}{k}}(\varepsilon n)^{-1/k}\right).$$

The privacy of the choice of d' is guaranteed as we only use the private covariance matrix.

7. Near-optimal accuracy bound with additional assumptions when $d' = 1$

Our Proposition 8 is not applicable to the case $d' = 1$ because the projection error in Theorem 3 only has bound $O((\varepsilon n)^{-\frac{1}{2}})$, which does not match with the optimal synthetic data accuracy bound in [12] and [31]. We are able to improve the accuracy bound with an additional dependence on $\sigma_1(\mathbf{M})$ as follows:

THEOREM 9. When $d' = 1$, consider Algorithm 1 with input data $\mathbf{X}$, output data $\mathbf{Y}$ and the subroutine PMM in Algorithm 4. Let $\mathbf{M}$ be the covariance matrix defines as (1.2). Assume $\sigma_1(\mathbf{M}) > 0$ and $n \geq 1/\varepsilon$,

then

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \lesssim \sqrt{\sum_{i>1} \sigma_i(\mathbf{M})} + \frac{d^3}{\sqrt{\sigma_1(\mathbf{M})\varepsilon n}} + \frac{\sqrt{d}\log^2(\varepsilon n)}{\varepsilon n}.$$

We start with the following lemma based on the Davis–Kahan theorem [59].

LEMMA 5. Let $\mathbf{X}$ be a $d \times n$ matrix and $\mathbf{A}$ be an $d \times d$ Hermitian matrix. Let $\mathbf{M} = \frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$, with the SVD

$$\mathbf{M} = \sum_{j=1}^{d} \sigma_j v_j v_j^\mathsf{T},$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d$ are the singular values of $\mathbf{M}$ and $v_1, \ldots, v_d$ are corresponding orthonormal eigenvectors. Let $\widehat{\mathbf{M}} = \frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T} + \mathbf{A}$ with orthonormal eigenvectors $\widehat{v}_1, \ldots, \widehat{v}_d$, where $\widehat{v}_1$ corresponds to the top singular value of $\widehat{\mathbf{M}}$. When there exists a spectral gap $\sigma_1 - \sigma_2 = \delta > 0$, we have

$$\frac{1}{n}\|\mathbf{X} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\mathbf{X}\|_F^2 \leq 2\sum_{i>1}\sigma_i + \frac{8d'^2}{n\delta^2}\|\mathbf{A}\|^2\|\mathbf{X}\|_F^2.$$

Proof. We have that

$$\begin{aligned}
\frac{1}{n}\|\mathbf{X} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\mathbf{X}\|_F^2 &= \frac{1}{n}\|\mathbf{X} - v_1v_1^\mathsf{T}\mathbf{X} + v_1v_1^\mathsf{T}\mathbf{X} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\mathbf{X}\|_F^2 \\
&\leq \frac{2}{n}\left(\|\mathbf{X} - v_1v_1^\mathsf{T}\mathbf{X}\|_F^2 + \|v_1v_1^\mathsf{T}\mathbf{X} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\mathbf{X}\|_F^2\right) \\
&= 2\sum_{i>1}\sigma_i + \frac{2}{n}\|\left(v_1v_1^\mathsf{T} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\right)\mathbf{X}\|_F^2 \\
&\leq 2\sum_{i>1}\sigma_i + \frac{2}{n}\|v_1v_1^\mathsf{T} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\|^2\|\mathbf{X}\|_F^2. \qquad (7.1)
\end{aligned}$$

To bound the operator norm distance between the two projections, we will need the Davis-Kahan Theorem in the perturbation theory. For the angle $\Theta(v_1, \widehat{v}_1)$ between the vectors v_1 and $\widehat{v}_1$, applying [59, Corollary 1], we have

$$\|v_1v_1^\mathsf{T} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\| = \sin\Theta(v_1, \widehat{v}_1) \leq \frac{2\|\mathbf{M} - \widehat{\mathbf{M}}\|}{\sigma_1 - \sigma_2} \leq \frac{2\|\mathbf{A}\|}{\delta}.$$

Therefore, when the spectral gap exists ($\delta > 0$),

$$\frac{1}{n}\|\mathbf{X} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\mathbf{X}\|_F^2 \leq 2\sum_{i>1}\sigma_i + \frac{8}{n\delta^2}\|\mathbf{A}\|^2\|\mathbf{X}\|_F^2.$$

This finishes the proof. □

Compared to Lemma 3, with the extra spectral gap assumption, the dependence on $\mathbf{A}$ in the upper bound changes from $\|\mathbf{A}\|$ to $\|\mathbf{A}\|^2$. A similar phenomenon, called *global and local bounds*, was observed in [48, Proposition 2.2]. With Lemma 5, we are able to improve the accuracy rate for the noisy projection step as follows:

PROPOSITION 10. Let $\sigma_1 \geq \cdots \geq \sigma_d \geq 0$ be the singular values of $\mathbf{M}$ defined in (3.6). When $d' = 1$, assume that $\sigma_1 > 0$ and $n \geq 1/\varepsilon$. For the output $\widehat{\mathbf{X}}$ in Algorithm 3, we have

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) \leq \left(\mathbb{E}\, W_2^2(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}})\right)^{1/2} \lesssim \sqrt{\sum_{i>1}\sigma_i} + \frac{d^3}{\sqrt{\sigma_1 \varepsilon n}},$$

Proof. Similar to the proof of Theorem 3, we can define $Z_i = X_i - \overline{X}$ and deduce that

$$\frac{1}{n}\mathbf{Z}\mathbf{Z}^\mathsf{T} = \frac{n-1}{n}\mathbf{M},$$
$$\frac{1}{n}\|\mathbf{Z}\|_F^2 = \frac{n-1}{n}\operatorname{tr}(\mathbf{M}),$$

and

$$W_2^2(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) = \frac{2}{n}\|\mathbf{Z} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\mathbf{Z}\|_F^2 + 2\|\lambda\|_2^2.$$

By the inequality $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$ for $x, y \geq 0$,

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) \leq \mathbb{E}\left[\frac{2}{n}\|\mathbf{Z} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\mathbf{Z}\|_F^2\right]^{1/2} + \sqrt{2}\,\mathbb{E}\,\|\lambda\|_2.$$

Let $\delta = \sigma_1 - \sigma_2$. Next, we will discuss two cases for the value of δ.

Case 1: When $\delta = \sigma_1 - \sigma_2 \leq \frac{1}{2}\sigma_1$, we have $\sigma_1 \leq 2\sigma_2$ and

$$\operatorname{tr}(\mathbf{M}) = \sigma_1 + \cdots + \sigma_d \leq 3\sum_{i>1}\sigma_i.$$

As any projection map has spectral norm 1, we have $\|v_1 v_1^\mathsf{T} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\| \leq 2$. Applying (7.1), we have

$$\begin{aligned}\frac{1}{n}\|\mathbf{Z} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\mathbf{Z}\|_F^2 &\leq 2\sum_{i>1}\sigma_i + \frac{2}{n}\|v_1 v_1^\mathsf{T} - \widehat{v}_1\widehat{v}_1^\mathsf{T}\|^2\|\mathbf{Z}\|_F^2 \\ &\leq 2\sum_{i>1}\sigma_i + \frac{8}{n}\|\mathbf{Z}\|_F^2 \\ &\leq 2\sum_{i>1}\sigma_i + 8\operatorname{tr}(\mathbf{M}) \leq 26\sum_{i>1}\sigma_i.\end{aligned}$$

Hence

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}-\bar{X}\mathbf{1}^\intercal}, \mu_{\widehat{\mathbf{X}}}) \lesssim \sqrt{\sum_{i>1}\sigma_i} + \mathbb{E}\,\|\lambda\|_2 \lesssim \sqrt{\sum_{i>1}\sigma_i} + \frac{\sqrt{d}}{\varepsilon n}. \tag{7.2}$$

Case 2: When $\delta \geq \frac{1}{2}\sigma_1$, we have

$$\operatorname{tr}(\mathbf{M}) \leq d\sigma_1 \leq \frac{4d\delta^2}{\sigma_1}.$$

For any fixed δ, by Lemma 5,

$$\begin{aligned}\frac{1}{n}\|\mathbf{Z} - \widehat{v}_1\widehat{v}_1^\intercal\mathbf{Z}\|_F^2 &\leq 2\sum_{i>1}\sigma_i + \frac{8}{n\delta^2}\|\mathbf{A}\|^2\|\mathbf{Z}\|_F^2 \\ &\leq 2\sum_{i>1}\sigma_i + \frac{8}{\delta^2}\|\mathbf{A}\|^2\operatorname{tr}(\mathbf{M}) \\ &\leq 2\sum_{i>1}\sigma_i + \frac{32d}{\sigma_1}\|\mathbf{A}\|^2.\end{aligned}$$

So we have the Wasserstein distance bound

$$\begin{aligned}\mathbb{E}\, W_1(\mu_{\mathbf{X}-\bar{X}\mathbf{1}^\intercal}, \mu_{\widehat{\mathbf{X}}}) &\leq \sqrt{2\sum_{i>1}\sigma_i} + \sqrt{\frac{32d}{\sigma_1}}\,\mathbb{E}\,\|\mathbf{A}\| + \sqrt{2}\,\mathbb{E}\,\|\lambda\|_2 \\ &\leq \sqrt{2\sum_{i>1}\sigma_i} + \sqrt{\frac{32d}{\sigma_1}}\frac{d^{2.5}}{\varepsilon n} + \frac{\sqrt{2d}}{\varepsilon n} \\ &\leq \sqrt{2\sum_{i>1}\sigma_i} + \frac{Cd^3}{\sqrt{\sigma_1}\varepsilon n}.\end{aligned} \tag{7.3}$$

From (3.6),

$$\sigma_1 = \|M\| \leq \|M\|_F \leq \frac{n}{n-1}d \leq 2d.$$

Combining the two cases (7.2) and (7.3), we deduce the result. □

Proof of Theorem 9 Following the steps in the proof of Theorem 3, we obtain

$$\begin{aligned}\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathscr{A}(\mathbf{X})}) &\le 2\,\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}'+(\overline{X}+\lambda')\mathbf{1}^\mathsf{T}}) \\ &\le 2\,\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) + 2\,\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) + 2\,\mathbb{E}\,\|\lambda'\|_\infty \\ &\lesssim \sqrt{\sum_{i>1}\sigma_i} + \frac{d'd^3}{\sqrt{\sigma_1}\varepsilon n} + \frac{\sqrt{d}\log^2(\varepsilon n)}{\varepsilon n} + \frac{2C\log d}{\varepsilon n} \\ &\lesssim \sqrt{\sum_{i>1}\sigma_i} + \frac{d'd^3}{\sqrt{\sigma_1}\varepsilon n} + \frac{\sqrt{d}\log^2(\varepsilon n)}{\varepsilon n},\end{aligned}$$

where for the second inequality, we apply the bound from [31, Theorem 1.1] for the second term, and we use (5.4) for the third term. □

8. Conclusion

In this paper, we provide a DP algorithm to generate synthetic data, which closely approximates the true data in the hypercube $[0,1]^d$ under 1-Wasserstein distance. Moreover, when the true data lies in a d'-dimensional affine subspace, we improve the accuracy guarantees in [31] and circumvents the curse of dimensionality by generating a synthetic dataset close to the affine subspace.

It remains open to determine the optimal dependence on d in the accuracy bound in Proposition 8 and whether the third term in (5.1) is needed. Our analysis of private PCA works without using the classical Davis-Kahan inequality that requires a spectral gap on the dataset. However, to approximate a dataset close to a line ($d' = 1$), additional assumptions are needed in our analysis to achieve the near-optimal accuracy rate, see Section 7. It is an interesting problem to achieve an optimal rate without the dependence on $\sigma_1(\mathbf{M})$ when $d' = 1$.

Our Algorithm 1 only outputs synthetic data with a low-dimensional linear structure, and its analysis heavily relies on linear algebra tools. For original datasets from a d'-dimensional linear subspace, we improve the accuracy rate from $(\varepsilon n)^{-1/(d'+1)}$ in [19] to $(\varepsilon n)^{-1/d'}$. It is also interesting to provide algorithms with optimal accuracy rates for datasets from general low-dimensional manifolds beyond the linear setting.

Funding

DOE-SC0023490, NIH R01HL16351, NSF DMS-2027248 and NSF DMS-2208356 (to T.S.); NSF DMS-1954233, NSF DMS-2027299, U.S. Army 76649-CS and NSF+Simons Research Collaborations on the Mathematical and Scientific Foundations of Deep Learning (to R.V.); NSF-Simons Research Collaborations on the Mathematical and Scientific Foundations of Deep Learning and an AMS-Simons Travel Grant (to Y.Z.).

Data Availability Statement

No new data were generated or analysed in support of this review.

References

1. Abowd, J., Ashmead, R., Simson, G., Kifer, D., Leclerc, P., Machanavajjhala, A. & Sexton, W. (2019) *Census Topdown: Differentially Private Data, Incremental Schemas, and Consistency With Public Knowledge*. US Census Bureau.
2. Abowd, J. M., Ashmead, R., Cumings-Menon, R., Garfinkel, S., Heineck, M., Heiss, C., Johns, R., Kifer, D., Leclerc, P., Machanavajjhala, A., Moran, B., Sexton, W., Spence, M. & Zhuravlev, P. (2022) The 2020 census disclosure avoidance system topdown algorithm. *Harvard Data Sci. Rev.*, **2**, (Special Issue 2).
3. Achlioptas, D. & McSherry, F. (2001). Fast computation of low-rank matrix approximation. In *Proceedings of the thirty-P third annual ACM symposium on Theory of computing*, pp. 611–618. ACM.
4. Amin, K., Dick, T., Kulesza, A., Munoz, A. & Vassilvitskii, S. (2019) Differentially private covariance estimation. *Adv. Neural Inf. Process. Syst.*, **32**.
5. Arora, R., Upadhyay, J. & Braverman, V. (2018) Differentially private robust low-rank approximation. *Adv. Neural Inf. Process. Syst.*, **31**.
6. Balog, M., Tolstikhin, I. & Schölkopf, B. (2018) Differentially private database release via kernel mean embeddings. In *International Conference on Machine Learning*, pp. 414–422. PMLR.
7. Bellovin, S. M., Dutta, P. K. & Reitinger, N. (2019) Privacy and synthetic datasets. *Stan. Tech. L. Rev.*, **22**, 1.
8. Bhatia, R. (2013) *Matrix Analysis*, vol. **169**. Springer Science & Business Media.
9. Blum, A., Ligett, K. & Roth, A. (2013) A learning theory approach to noninteractive database privacy. *J. ACM*, **60**, 1–25.
10. Boedihardjo, M., Strohmer, T. & Vershynin, R. (2022) Private sampling: a noiseless approach for generating differentially private synthetic data. *SIAM J. Math. Data Sci.*, **4**, 1082–1115.
11. Boedihardjo, M., Strohmer, T. & Vershynin, R. (2024a) Covariance's loss is privacy's gain: computationally efficient, private and accurate synthetic data. *Found. Comput. Math.*, **24**, 179–226.
12. Boedihardjo, M., Strohmer, T. & Vershynin, R. (2024b) Private measures, random walks, and synthetic data. *Probab. Theory Related Fields*, **189**, 1–43.
13. Bubeck, S. & Sellke, M. (2021) A universal law of robustness via isoperimetry. *Adv. Neural Inf. Process. Syst.*, **34**, 28811–28822.
14. Chaudhuri, K., Monteleoni, C. & Sarwate, A. D. (2011) Differentially private empirical risk minimization. *J. Mach. Learn. Res.*, **12**, 1069–1109.
15. Chaudhuri, K., Sarwate, A. D. & Sinha, K. (2013) A near-optimal algorithm for differentially-private principal components. *J. Mach. Learn. Res.*, **14**.
16. Dai, G., Su, Z. & Wang, H. (2024) Tail bounds on the spectral norm of sub-exponential random matrices. *Random Matrices: Theor. Appl.*, **13**, 2350013.
17. Davis, C. & Kahan, W. M. (1970) The rotation of eigenvectors by a perturbation. III. *SIAM J. Numer. Anal.*, **7**, 1–46.
18. Dong, W., Liang, Y. & Yi, K. (2022) Differentially private covariance revisited. *Adv. Neural Inf. Process. Syst.*, **35**, 850–861.
19. Donhauser, K., Lokna, J., Sanyal, A., Boedihardjo, M., Hönig, R. & Yang, F. (2024) Certified private data release for sparse Lipschitz functions. In *International Conference on Artificial Intelligence and Statistics*, pp. 1396–1404. PMLR.
20. Dwork, C. & Roth, A. (2014) The algorithmic foundations of differential privacy. *Found. Trends. Theor. Comput. Sci.*, **9**, 211–407.
21. Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I. & Naor, M. (2006) Our data, ourselves: privacy via distributed noise generation. In *Advances in Cryptology-EUROCRYPT 2006: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28–June 1, 2006. Proceedings 25*, pp. 486–503. Springer.

22. Dwork, C., Naor, M., Reingold, O., Rothblum, G. N. & Vadhan, S. (2009) On the complexity of differentially private data release: efficient algorithms and hardness results. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pp. 381–390.
23. Dwork, C., Talwar, K., Thakurta, A. & Zhang, L. (2014) Analyze Gauss: optimal bounds for privacy-preserving principal component analysis. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pp. 11–20.
24. Dwork, C., Nikolov, A. & Talwar, K. (2015) Efficient algorithms for privately releasing marginals via convex relaxations. *Discrete Comput. Geom.*, **53**, 650–673.
25. Dwork, C., Kohli, N. & Mulligan, D. (2019) Differential privacy in practice: expose your epsilons! *J. Priv. Confidentiality*, **9**.
26. Feige, U. & Ofek, E. (2005) Spectral techniques applied to sparse random graphs. *Random Struct. Algorithms*, **27**, 251–275.
27. Harder, F., Adamczewski, K. & Park, M. (2021) Dp-merf: Differentially private mean embeddings with random features for practical privacy-preserving data generation. In *International conference on artificial intelligence and statistics*, pp. 1819–1827. PMLR.
28. Hardt, M. & Price, E. (2014) The noisy power method: a meta algorithm with applications. *Adv. Neural Inf. Process. Syst.*, **27**.
29. Hardt, M. & Roth, A. (2013) Beyond worst-case analysis in private singular vector computation. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pp. 331–340.
30. Hardt, M., Ligett, K. & McSherry, F. (2012) A simple and practical algorithm for differentially private data release. *Adv. Neural Inf. Process. Syst.*, **25**.
31. He, Y., Vershynin, R. & Zhu, Y. (2023) Algorithmically effective differentially private synthetic data. In G. Neu and L. Rosasco, eds, *Proceedings of Thirty Sixth Conference on Learning Theory, volume 195 of Proceedings of Machine Learning Research*, pp. 3941–3968. PMLR, 12–15 Jul.
32. Imtiaz, H. & Sarwate, A. D. (2016) Symmetric matrix perturbation for differentially-private principal component analysis. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2339–2343. IEEE.
33. Inusah, S. & Kozubowski, T. J. (2006) A discrete analogue of the Laplace distribution. *J. Statist. Plann. Inference*, **136**, 1090–1102.
34. Jain, P., Jin, C., Kakade, S. M., Netrapalli, P. & Sidford, A. (2016) Streaming PCA: matching matrix Bernstein and near-optimal finite sample guarantees for Oja's algorithm. In *Conference on learning theory*, pp. 1147–1164. PMLR.
35. Jiang, W., Xie, C. & Zhang, Z. (2016) Wishart mechanism for differentially private principal components analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. **30**.
36. Jiang, X., Ji, Z., Wang, S., Mohammed, N., Cheng, S. & Ohno-Machado, L. (2013) Differential-private data publishing through component analysis. *Trans. Data Priv.*, **6**, 19.
37. Kamath, G., Li, J., Ullman, V. & Ullman, J. (2019) Privately learning high-dimensional distributions. In *Conference on Learning Theory*, pp. 1853–1902. PMLR.
38. Kapralov, M. & Talwar, K. (2013) On differentially private low rank approximation. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1395–1414. SIAM.
39. Kovalev, L. V. (2022) Lipschitz clustering in metric spaces. *J. Geom. Anal.*, **32**, 188.
40. Kreacic, E., Nouri, N., Potluru, V. K., Balch, T. & Veloso, M. (2023) Differentially private synthetic data using kd-trees. In *The 39th Conference on Uncertainty in Artificial Intelligence*.
41. Li, X., Wang, S. & Cai, Y. (2019) Tutorial: complexity analysis of singular value decomposition and its variants. arXiv preprint arXiv:1906.12085.
42. Liu, X., Kong, W., Kakade, S. & Oh, S. (2021) Robust and differentially private mean estimation. *Adv. Neural Inf. Process. Syst.*, **34**, 3887–3901.
43. Liu, X., Kong, W., Jain, P. & Oh, S. (2022a) DP-PCA: statistically optimal and differentially private pca. *Adv. Neural Inf. Process. Syst.*, **35**, 29929–29943.

44. Liu, X., Kong, W. & Oh, S. (2022b) Differential privacy and robust statistics in high dimensions. In *Conference on Learning Theory*, pp. 1167–1246. PMLR.
45. Mangoubi, O. & Vishnoi, N. (2022) Re-analyze Gauss: bounds for private matrix approximation via Dyson Brownian motion. *Adv. Neural Inf. Process. Syst.*, **35**, 38585–38599.
46. Meunier, L., Delattre, B. J., Araujo, A. & Allauzen, A. (2022) A dynamical system perspective for Lipschitz neural networks. In *International Conference on Machine Learning*, pp. 15484–15500. PMLR.
47. Oja, E. (1982) Simplified neuron model as a principal component analyzer. *J. Math. Biol.*, **15**, 267–273.
48. Reiss, M. & Wahl, M. (2020) Nonasymptotic upper bounds for the reconstruction error of PCA. *Ann. Statist.*, **48**, 1098–1123.
49. Singhal, V. & Steinke, T. (2021) Privately learning subspaces. *Adv. Neural Inf. Process. Syst.*, **34**, 1312–1324.
50. Thaler, J., Ullman, J. & Vadhan, S. (2012) Faster algorithms for privately releasing marginals. In *Automata, Languages, and Programming: 39th International Colloquium, ICALP 2012, Warwick, UK, July 9–13, 2012, Proceedings, Part I 39*, pp. 810–821. Springer.
51. Tsfadia, E. (2024) On differentially private subspace estimation in a distribution-free setting. arXiv preprint arXiv:2402.06465.
52. Ullman, J. & Vadhan, S. (2011) PCPs and the hardness of generating private synthetic data. In *Theory of Cryptography: 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, March 28–30, 2011. Proceedings 8*, pp. 400–416. Springer.
53. Vietri, G., Archambeau, C., Aydore, S., Brown, W., Kearns, M., Roth, A., Siva, A., Tang, S. & Wu, S. (2022) Private synthetic data for multitask learning and marginal queries. *Adv. Neural Inf. Process. Syst,* **35**, 18282–18295.
54. Villani, C. (2009) *Optimal Transport: Old and New*, vol. **338**. Springer.
55. von Luxburg, U. & Bousquet, O. (2004) Distance-based classification with Lipschitz functions. *J. Mach. Learn. Res.*, **5**, 669–695.
56. Wang, Z., Jin, C., Fan, K., Zhang, J., Huang, J., Zhong, Y. & Wang, L. (2016) Differentially private data releasing for smooth queries. *J. Mach. Learn. Res.*, **17**, 1779–1820.
57. Wasserman, L. & Zhou, S. (2010) A statistical framework for differential privacy. *J. Am. Statist. Assoc.*, **105**, 375–389.
58. Yang, Y., Adamczewski, K., Sutherland, D. J., Li, X. & Park, M. (2023) Differentially private neural tangent kernels for privacy-preserving data generation. arXiv preprint arXiv:2303.01687.
59. Yu, Y., Wang, T. & Samworth, R. J. (2015) A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, **102**, 315–323.
60. Zhou, S., Ligett, K. & Wasserman, L. (2009) Differential privacy with compression. In *2009 IEEE International Symposium on Information Theory*, pp. 2718–2722. IEEE.