DIFFERENTIALLY PRIVATE LOW-DIMENSIONAL REPRESENTATION OF HIGH-DIMENSIONAL DATA

YIYUN HE, THOMAS STROHMER, ROMAN VERSHYNIN, AND YIZHE ZHU

Abstract

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. Different from the standard perturbation analysis using the Davis-Kahan theorem, our analysis of private PCA works without assuming the spectral gap for the sample covariance matrix.

1. Introduction

As data sharing is increasingly locking horns with data privacy concerns, privacy-preserving data analysis has emerged as a challenging task with far-reaching impact. Differential privacy (DP) has emerged as a de facto standard for implementing privacy in various applications [18]. For instance, DP has been adopted by several technology companies [16] and has also been used in connection with the release of Census 2020 data [2, 1, 3].

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 1.1 (Differential Privacy [18]). *A randomized algorithm $\mathcal{M}$ gives ε-differential privacy if for any neighboring datasets D and D' and any measurable subset $S \subseteq \mathrm{range}(\mathcal{M})$, we have*

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

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

However, utility guarantees for DP are usually provided only for a fixed, a priori specified set of queries. Moreover, utility guarantees are typically not available for more complex—but very common—machine learning tasks such as clustering or classification.

Hence, it has been frequently recommended that differential privacy may be combined with synthetic data to achieve more flexibility in private data sharing [21, 5]. Synthetic datasets are generated from existing datasets and maintain the statistical properties of the original dataset. Ideally, synthetic data contain no protected information; hence the datasets can be shared freely among investigators in academia or industry, without security and privacy concerns.

Yet, the numerically efficient construction of accurate differentially private synthetic data is a rather challenging task. Most research on private synthetic data has been concerned with counting queries, range queries, or k-dimensional marginals, see e.g. [21, 37, 7, 38, 17, 36, 10]. Notable exceptions are [41, 9, 14]. Specifically, [9] provides utility guarantees with respect to the 1-Wasserstein distance. Invoking the Kantorovich-Rubinstein duality theorem, the choice of the 1-Wasserstein distance to quantify accuracy ensures that all Lipschitz statistics are preserved uniformly. This provides data analysts with a vastly increased toolbox of machine learning methods for which one can expect similar outcomes for the original data and the synthetic data, respectively.

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 [9] 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 [24] 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 [9, 24] are only useful when d, the dimensionality of the data, is small (or if n is exponentially larger than d). In other words, we are facing the curse of dimensionality.

In [14], 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. However, the optimization step in their algorithm exhibits exponential time complexity. In this paper, we present an efficient algorithm that does not rely on any assumptions about the true data. Notably, 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 dimensionality 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. Unlike classical perturbation analysis that utilizes the Davis-Kahan theorem [13] in the literature [11, 19], our accuracy analysis of private PCA works without assuming the spectral gap for the sample covariance matrix.

Notation. In this paper, we work with data in the Euclidean space $\mathbb{R}^d$. For convenience, the data matrix $\mathbf{X} = [X_1, \dots, X_n] \in \mathbb{R}^{d\times n}$ also indicates the dataset $(X_1, \dots, X_n)$. We use $\mathbf{A}$ to denote a matrix and v, X as vectors. $\|\cdot\|_F$ is the Frobenius norm and $\|\cdot\|$ is the operator norm of a matrix, respectively. Two sequences a_n, b_n satisfies $a_n \lesssim b_n$ if $a_n \le C b_n$ 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 an informal 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. In Section 2, we provide useful lemmas and definitions. Next, we consider the Algorithm 1 step by step. In Section 3, we discuss private PCA and noisy projection. In Section 4, we apply synthetic data algorithms from [24] to the specific cases on the lower dimensional spaces. The precise privacy and accuracy guarantee of Algorithm 1 is summarized in Section 5. Finally, since the case $d' = 1$ is not covered in Theorem 1.2, we discuss additional results under stronger assumptions in Section 6. All the proofs can be found in the Supplementary Material.

1.1. Main results.

The problem of generating private synthetic data can be defined as follows. Let (Ω, ρ) be a metric space. Consider a dataset $\mathbf{X} = [X_1, \dots, X_n] \in \Omega^n$. Our goal is to construct an efficient differentially private randomized algorithm that outputs synthetic data $\mathbf{Y} = [Y_1, \dots, Y_n] \in \Omega^m$ 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. 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 an efficient algorithm for this task. It can be summarized in the following 4 steps:

1. Construct a private sample covariance matrix $\widehat{\mathbf{M}}$. The private sample covariance is constructed by adding a Laplacian random matrix to a centered sample covariance matrix $\mathbf{M}$. 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 following methods in [24]. This is summarized in Algorithms 4 and 5.
4. Add a private mean vector back to the dataset and metrically project back them to the hypercube $[0,1]^d$. Output the resulting dataset $\mathbf{Y}$. This step is summarized in the last two parts of Algorithm 1.

The privacy and accuracy guarantees of Algorithm 1 are stated in the next informal theorem. More detailed and precise statements are given in Section 5.

Theorem 1.2. *Let $\Omega = [0,1]^d$ equipped with ℓ^∞ metric and $\mathbf{X} = [X_1, \dots, X_n] \in \Omega^n$ be a dataset. For any $2 \le d' \le d$, Algorithm 1 outputs an ε-differentially private synthetic dataset $\mathbf{Y} = [Y_1, \dots, Y_m] \in \Omega^m$ for some $m \ge 1$ in polynomial time such that*

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \lesssim_d \sqrt{\sum_{i>d'} \sigma_i(\mathbf{M})} + (\varepsilon n)^{-1/d'}, \tag{1.2}$$

where $\lesssim_d$ means the right hand side of (1.2) hides factors that are polynomial in d, and $\sigma_i(\mathbf{M})$ is the i-th eigenvalue value of the sample covariance matrix of $\mathbf{X}$.

Algorithm 1 Low-dimensional Synthetic Data

Input: True data matrix $\mathbf{X} = [X_1, \dots, X_n]$, $X_i \in [0,1]^d$ and privacy parameter ε.

Private linear projection: Choose a parameter d' based on a private covariance matrix constructed in Algorithm 2. Use Algorithm 3 to project $\mathbf{X}$ onto to a d'-dimensional linear subspace with privacy parameter $\varepsilon/2$.

Low-dimensional synthetic data: Use subroutine in Section 4 to generate $\varepsilon/4$-private synthetic data $\mathbf{X}'$ depending on $d' = 2$ or $d' \ge 3$.

Adding the private mean vector: Let $\overline{X}$ be the mean value of the dataset and $X_i'' = X_i' + \overline{X} + \lambda'$, where λ' is a random vector with i.i.d. components of $\mathrm{Lap}(4/(\varepsilon n))$.

Metric projection: Project each data to the nearest point in $[0,1]^d$. Define a function $f : \mathbb{R} \to [0,1]$ 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. Take $\mathbf{Y} = [f(X_1''), \dots, f(X_n'')]$.

Output: Synthetic data $\mathbf{Y}$.

Optimality. The accuracy rate in (1.2) is optimal up to a polynomial factor in d. The first term matches the error from the best rank-d' approximation of the covariance matrix $\mathbf{M}$, and the second term matches the lower bound in [9, Corollary 9.3] for generating d'-dimensional synthetic data in $[0,1]^{d'}$.

Improved accuracy if X is low-dimensional. When the original dataset $\mathbf{X}$ lies in an affine d'-dimensional subspace, it implies $\sigma_i(\mathbf{M}) = 0$ for $i > d'$ and $\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \lesssim_d (\varepsilon n)^{-1/d'}$. This is an improvement from the accuracy rate $O((\varepsilon n)^{-1/d})$ for unstructured data in $[0,1]^d$ in [9, 24], which overcomes the curse of high dimensionality.

$\mathbf{Y}$ is a low-dimensional representation of $\mathbf{X}$. The synthetic dataset $\mathbf{Y}$ is close to a d'-dimensional subspace under the 1-Wasserstein distance, as shown in Proposition 4.5.

Private and adaptive choice of d'. In fact, it is possible to choose the value of d' adaptively yet privately based on the private sample covariance matrix in Algorithm 2. A private and near-optimal choice of d' is given in Remark 5.4.

Algorithm efficiency. The *private linear projection* step has a running time $O(d^2 n)$ using truncated SVD [30]. The *low-dimensional synthetic data* subroutine has a running time polynomial in n for $d' \geq 3$ and linear in n when $d' = 2$ [24]. Therefore, the overall running time for Algorithm 1 is linear in n when $d' = 2$ and is polynomial in n when $d' \geq 3$. Although sub-optimal in the accuracy guarantee in terms of the dependence on d, one can run Algorithm 1 in linear time by choosing PMM (Algorithm 4) in the subroutine for all $d' \geq 2$.

1.2. Comparison to previous results.

Private PCA is a commonly used technique for differentially private low-rank approximation of the original dataset by adding noise to the true sample covariance matrix[1]. Adding a Gaussian random matrix for this task has drawn significant interest [33, 11, 25, 19], but they only permit a weaker version of (ε, δ)-differential privacy. Alternatively, to achieve ε-differential privacy, adding Laplacian random matrix perturbation to the sample covariance matrix is also widely used in private PCA [27, 28, 43]. Instead of adding independent noise, the method of exponential mechanism is also broadly studied [29], where the private covariance matrix is sampled from a certain distribution, such as the matrix Bingham distribution [11] or the Wishart distribution [27]. Besides working with the sample covariance matrix, another approach, called streaming PCA [34, 26], which aims to compute the top eigenvector, can also be performed privately [22, 31].

The standard output of private PCA is 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 $\mathbf{V}_{d'}$ [19, 23, 33, 31]. To prove a utility guarantee, the standard tool is the Davis-Kahan Theorem [6, 40, 42], which assumes that the sample covariance matrix (centered or non-centered) has a spectral gap [11, 19, 22, 27, 31]. Alternatively, using the projection error to evaluate accuracy is independent of the spectral gap [29, 32]. In our application of private PCA, $\widehat{\mathbf{V}}_{d'}$ is not our final output and we project $\mathbf{X}$ onto $\widehat{\mathbf{V}}_{d'}$ afterwards. We instead directly bound the Wasserstein distance between the projected dataset and $\mathbf{X}$, which avoids the subspace perturbation analysis, leading to a final accuracy bound independent of the spectral gap (see Lemma 3.2).

We emphasize that working with the centered sample covariance matrix in the private PCA step is essential to obtain the near-optimal accuracy in Theorem 1.2, as the centered sample covariance matrix is of rank d' if $\mathbf{X}$ lies in a d'-dimensional affine space, but $\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$ has rank $(d'+1)$. The

[1] Note that a common choice of the sample covariance matrix for PCA is $\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$, which is different from the centered sample covariance matrix we define in (3.1).

centering step improves the accuracy rate from $(\varepsilon n)^{-1/(d'+1)}$ to $(\varepsilon n)^{-1/d'}$. However, it comes with an additional task to protect the privacy of mean vectors (see the third step in Algotihm 1 and Algorithm 3).

2. Preliminaries

In this paper, we use Definition 1.1 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.

Differentially private algorithms have a useful property that their sequential composition is also differentially private [18, Theorem 3.16]. Moreover, the following result about *adaptive composition* indicates that algorithms in a sequential composition may use the outputs in the previous steps:

Lemma 2.1. [15, Theorem 1] *Suppose a randomized algorithm $\mathcal{M}_1(x) : \Omega^n \to \mathcal{R}_1$ is ε_1-differentially private, and $\mathcal{M}_2(x, y) : \Omega^n \times \mathcal{R}_1 \to \mathcal{R}_2$ is ε_2-differentially private with respect to the first component for any fixed y. Then the sequential composition*

$$x \mapsto (\mathcal{M}_1(x), \mathcal{M}_2(x, \mathcal{M}_1(x)))$$

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

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

Definition 2.2 (p-Wasserstein distance)**.** *Consider a metric space (Ω, ρ). The p*-Wasserstein distance *(see e.g., [39] 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.

3. Private linear projection

3.1. Private centered sample covariance matrix.

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 $\mathbf{X} = [X_1, \dots, X_n]$, where $X_1, \dots, X_n \in \mathbb{R}^d$. The rank of the sample covariance matrix $\frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T}$ measures the dimension of the *linear subspace* spanned by $X_1, \dots, X_n$. If we subtract the mean vector and consider the *centered sample covariance matrix*

$$\mathbf{M} = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \overline{X})(X_i - \overline{X})^\mathsf{T}, \quad \text{where} \quad \overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i, \tag{3.1}$$

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:

Theorem 3.1. *Algorithm 2 is ε-differentially private.*

Algorithm 2 Private Covariance Matrix

```
Input: Matrix $\mathbf{X} = [X_1, \dots, X_n]$ where $X_i \in [0,1]^d$, and privacy parameter $\varepsilon$, variance parameter $\sigma = \frac{3d^2}{\varepsilon n}$.
  Computing the sample covariance matrix: Compute the mean $\overline{X} = \frac{1}{n}\sum_{i=1}^n X_i$ and the centered sample 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
      $\mathbf{A}_{ij} = \begin{cases} \lambda_{ij} & \text{if } i < j; \\ 2\lambda_{ii} & \text{if } i = j, \end{cases}$
Output: The noisy covariance matrix $\widehat{\mathbf{M}} = \mathbf{M} + \mathbf{A}$.
```

Algorithm 3 Noisy Projection

```
Input: True data matrix $\mathbf{X} = [X_1, \dots, X_n]$, $X_i \in [0,1]^d$, privacy parameter $\varepsilon$, and the output dimension $d'$.
  Private covariance matrix: Apply Algorithm 2 for to $\mathbf{X}$ with privacy parameter $\varepsilon$ to obtain a private covariance matrix $\widehat{\mathbf{M}}$.
  Singular value decomposition: Compute the truncated SVD $\widehat{\mathbf{M}} = \sum_{j=1}^{d'} \widehat{\sigma}_j \widehat{v}_j \widehat{v}_j^{\mathsf{T}}$, where $\widehat{\sigma}_1 \ge \widehat{\sigma}_2 \ge \cdots \ge \widehat{\sigma}_{d'}$ are the eigenvalues of $\mathbf{M}$ and $\widehat{\mathbf{V}}_{d'} = [\widehat{v}_1, \dots, \widehat{v}_{d'}]$ are corresponding orthonormal eigenvectors.
  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 $\mathrm{Lap}(1/(\varepsilon n))$. 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 $\widehat{v}_1, \dots, \widehat{v}_{d'}$. 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 $\widehat{\mathbf{X}} = [\widehat{X}_1 \dots \widehat{X}_n]$.
```

3.2. **Noisy projection.** The private sample 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.

Note that Algorithm 3 only guarantees private basis $\widehat{v}_1, \dots, \widehat{v}_{d'}$ for each $\widehat{X}_i$, but the coordinates of $\widehat{X}_i$ in terms of $\widehat{v}_1, \dots, \widehat{v}_{d'}$ are *not private*. For different choice of d', 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 2.1, 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 sample covariance matrix with additive noise. We first provide the following useful deterministic lemma.

Lemma 3.2 (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 the singular value decomposition (SVD)* $\mathbf{M} = \sum_{j=1}^d \sigma_j v_j v_j^{\mathsf{T}}$*, where* $\sigma_1 \ge$

$\sigma_2 \geq \cdots \geq \sigma_d$ are the eigenvalues of M and $v_1 \dots v_d$ are corresponding orthonormal eigenvectors. 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

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

Inequality (3.2) holds without any spectral gap assumption on $\mathbf{M}$. When $\widehat{\mathbf{M}}$ is an empirical sample covariance matrix, a similar bound without a spectral gap condition is derived in [35, Proposition 2.2]. In Lemma 3.2, we do not assume $\widehat{\mathbf{M}}$ is positive semidefinite. Lemma 3.2 has a similar flavor to [4, Theorem 5] in the context of low-rank matrix approximation of a rectangular matrix under perturbation, and our proof is inspired by Parseval's identity used in [4].

With Lemma 3.2, we derive Wasserstein distance bounds between the centered dataset $\mathbf{X} - \overline{X}\mathbf{1}^\mathsf{T}$ and the dataset $\widehat{\mathbf{X}}$.

Theorem 3.3. *For input data $\mathbf{X}$ and output data $\widehat{\mathbf{X}}$ in Algorithm 3, let $\mathbf{M}$ be the sample covariance matrix defined in (3.1). Then for an absolute constant $C > 0$,*

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

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

$$\left\|\widehat{X}_i\right\|_2 = \left\|X_i - \overline{X} - \lambda\right\|_2 \leq \sqrt{d} + \left\|\overline{X} + \lambda\right\|_2 =: R \tag{4.1}$$

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

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

where $\widehat{v}_1, \dots, \widehat{v}_{d'}$ are the first d' private principal components in Algorithm 3. Depending on whether $d' = 2$ or $d' \geq 3$, we apply two different algorithms from [24].

Remark 4.1 (R is private)**.** *R defined in (4.1) depends on the private mean. It is part of the input for the synthetic data subroutines, and it is independent of the randomness in the subroutines.*

4.1. $d' = 2$: private measure mechanism (PMM).

We use the private measure mechanism introduced in [24] with a specified bounded region. In this method, we need a binary partition structure for the region Ω'. However, for a high-dimensional ball like Ω', it is not easy to give an explicit binary partition to match its covering numbers. Instead, we can enlarge Ω' to a hypercube $[-R, R]^{d'}$ inside the linear subspace $\widehat{\mathbf{V}}_{d'}$. The privacy and accuracy guarantees follow from [24].

Proposition 4.2. *The subroutine Algorithm 4 is ε-differentially private. When $d' = 2$, with the input as the projected data $\widehat{\mathbf{X}}$ and the range Ω' with radius R, the algorithm has an accuracy bound*

$$\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) \leq CR(\varepsilon n)^{-1/2},$$

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

Algorithm 4 PMM Subroutine

Input: dataset $\widehat{\mathbf{X}} = (\widehat{X}_1, \dots, \widehat{X}_n)$ 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 $\sigma_j \sim \varepsilon^{-1} \cdot 2^{\frac{1}{2}(1-\frac{1}{d'})(r-j)}$. Enlarge the region Ω' into

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

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 Laplacian random variables with $\lambda \sim \text{Lap}(\sigma_{|\theta|})$, and $|\theta|$ is the length of the vector θ.

Consistency: Enforce consistency of $\{n'_\theta\}_{\theta \in \{0,1\}^{\le r}}$

Output: Synthetic data $\mathbf{X}'$ randomly sampled from $\{\Omega_\theta\}_{\theta \in \{0,1\}^r}$.

Remark 4.3 (PMM for $d' \ge 2$)**.** *For general $d' \ge 2$, PMM can still be applied, and the accuracy bound becomes $\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) \le CR(\varepsilon n)^{-1/d'}$. Compared to* (1.2)*, as $\mathbb{E}_\lambda R = \Theta(\sqrt{d})$, this accuracy bound is weaker by a factor of $\sqrt{d'}$. However, as shown in* [24]*, PMM has a running time linear in n, which is more efficient than PSMM given in Algorithm 5.*

Algorithm 5 PSMM Subroutine

Input: dataset $\widehat{\mathbf{X}} = (\widehat{X}_1, \dots, \widehat{X}_n)$ 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 $\delta \sim \sqrt{d/d'}(\varepsilon n)^{-1/d'}$. Find the lattice over the region:

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

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

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

where $\lambda_v \sim \text{Lap}(1/\varepsilon)$, $v \in L$ are i.i.d. random variables.

Synthetic probability measure: Use linear programming and find the closest probability measure to ν.

Output: Synthetic data corresponding to the probability measure.

4.2. $d' \ge 3$**: private signed measure mechanism (PSMM).** 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 [24]. We apply PSMM from [24] when the metric space is a ℓ_2-ball of radius R inside $\widehat{\mathbf{V}}_{d'}$ and the following privacy and accuracy guarantees hold:

Proposition 4.4. *The subroutine Algorithm 5 is ε-differentially private. And 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*

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

where the expectation is conditioned on R.

4.3. **Adding a private mean vector and metric projection.** After generating the synthetic data, since we shifted 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 affine subspace close to the original dataset $\mathbf{X}$. The output data vectors in $\mathbf{X}''$ (defined in Algorithm 1) after this step are not necessarily inside $[0,1]^d$. The metric projection step enforces all synthetic data to be inside $[0,1]^d$. This post-processing does not influence the privacy guarantee. After metric projection, dataset $\mathbf{Y}$ from the output of Algorithm 1 is close to an affine subspace, as shown in the next proposition.

Proposition 4.5 ($\mathbf{Y}$ is close to an affine subspace)**.** *The function $f : \mathbb{R}^d \to [0,1]^d$ 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:*

$$W_1(\mu_{\mathbf{Y}}, \mu_{\mathbf{X}''}) \le W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}),$$

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

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

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.

Theorem 5.1 (Privacy)**.** *Algorithm 1 is ε-differentially private.*

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

Theorem 5.2 (Accuracy)**.** *For $2 \le d' \le d$, the output data $\mathbf{Y}$ from Algorithm 1 with the input data $\mathbf{X}$ satisfies*

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

where $\mathbf{M}$ denotes the sample covariance matrix defines as (3.1).

There are three terms on the right-hand side of (5.1). The first term is due to the error from the rank-d' approximation of the covariance matrix $\mathbf{M}$. The second term is from the accuracy loss for private PCA after the perturbation from a random Laplacian matrix. It is not clear to us whether this error term is needed or optimal. The third term is from the accuracy loss when generating synthetic data in a low-dimensional subspace, and it matches the optimal accuracy rate for synthetic data on $[0,1]^{d'}$ in [9, 24] when $d' = d$. The scaling factor $\sqrt{d/d'}$ is from the private linear projection step, where we essentially project a d-dimensional ℓ_2-ball of radius $\sqrt{d}$ to a d'-dimensional ℓ_2-ball of radius $\sqrt{d}$. This can be seen more clearly in the proof of Proposition 4.4.

Remark 5.3 (Dependence on d)**.** *When $d' \ge 3$ and $d \le (d')^{-\frac{4}{3}}(\varepsilon n)^{\frac{2}{3}(1-2/d')}$, the error rate in* (5.1) *becomes $O\left(\sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'}\right)$. When $d' = 2$, we get a slightly worse error bound $O(d^{5/4}(\varepsilon n)^{-1/2})$.*

Remark 5.4 (Adaptive choice of d')**.** *Recall $\widehat{\mathbf{M}}$ from Algorithm 2. We can choose d' based on $\widehat{\mathbf{M}}$ privately such that*

$$d' := \operatorname*{arg\,min}_{2\leq k\leq d} \left(\sqrt{\sum_{i>d'} \sigma_i(\widehat{\mathbf{M}})} + \sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'} + \sqrt{\frac{d'd^{2.5}}{\varepsilon n}} \right). \tag{5.2}$$

Moreover, we have $\left|\sum_{i>d'} \sigma_i(\widehat{\mathbf{M}}) - \sum_{i>d'} \sigma_i(\mathbf{M})\right| \leq (d-d')\|\mathbf{A}\|$. Compared to the upper bound in Theorem 5.2, the choice of d' in (5.2) is near-optimal in Theorem 5.2 if $(d-d')\|\mathbf{A}\|$ is small.

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

Our Theorem 5.2 is not applicable to the case $d' = 1$ because the projection error in Theorem 3.3 only has bound $O((\varepsilon n)^{-\frac{1}{2}})$, which does not match with the optimal synthetic data accuracy bound in [9, 24]. We are able to improve the accuracy bound with an additional dependence on $\sigma_1(\mathbf{M})$ as follows:

Theorem 6.1. *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 sample covariance matrix defines as (3.1). Assume $\sigma_1(\mathbf{M}) > 0$, then*

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

7. Conclusion

In this paper, we provide a differentially private 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 [24] and circumvents the curse of dimensionality by generating a synthetic dataset close to the affine subspace.

For $d' \geq 2$, 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. It is an interesting problem to achieve the 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 [14] 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.

Acknowledgement

T.S. acknowledges support from NIH R01HL16351, NSF DMS-2027248, and NSF DMS-2208356. R.V. acknowledges support from 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. Y.Z. is partially supported by NSF-Simons Research Collaborations on the Mathematical and Scientific Foundations of Deep Learning.

REFERENCES

[1] John Abowd, Robert Ashmead, Garfinkel Simson, Daniel Kifer, Philip Leclerc, Ashwin Machanavajjhala, and William Sexton. Census topdown: Differentially private data, incremental schemas, and consistency with public knowledge. *US Census Bureau*, 2019.

[2] John M Abowd. The US Census Bureau adopts differential privacy. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2867–2867, 2018.

[3] John M Abowd, Robert Ashmead, Ryan Cumings-Menon, Simson Garfinkel, Micah Heineck, Christine Heiss, Robert Johns, Daniel Kifer, Philip Leclerc, Ashwin Machanavajjhala, et al. The 2020 census disclosure avoidance system topdown algorithm. *Harvard Data Science Review*, (Special Issue 2), 2022.

[4] Dimitris Achlioptas and Frank McSherry. Fast computation of low-rank matrix approximation. In *Proceedings of the thirty-P third annual ACM symposium on Theory of computing*, pages 611–618. ACM, 2001.

[5] Steven M Bellovin, Preetam K Dutta, and Nathan Reitinger. Privacy and synthetic datasets. *Stan. Tech. L. Rev.*, 22:1, 2019.

[6] Rajendra Bhatia. *Matrix analysis*, volume 169. Springer Science & Business Media, 2013.

[7] Avrim Blum, Katrina Ligett, and Aaron Roth. A learning theory approach to noninteractive database privacy. *Journal of the ACM (JACM)*, 60(2):1–25, 2013.

[8] March Boedihardjo, Thomas Strohmer, and Roman Vershynin. Covariance's loss is privacy's gain: Computationally efficient, private and accurate synthetic data. *Foundations of Computational Mathematics*, pages 1–48, 2022.

[9] March Boedihardjo, Thomas Strohmer, and Roman Vershynin. Private measures, random walks, and synthetic data. *arXiv preprint arXiv:2204.09167*, 2022.

[10] March Boedihardjo, Thomas Strohmer, and Roman Vershynin. Private sampling: a noiseless approach for generating differentially private synthetic data. *SIAM Journal on Mathematics of Data Science*, 4(3):1082–1115, 2022.

[11] Kamalika Chaudhuri, Anand D Sarwate, and Kaushik Sinha. A near-optimal algorithm for differentially-private principal components. *Journal of Machine Learning Research*, 14, 2013.

[12] Guozheng Dai, Zhonggen Su, and Hanchao Wang. Tail bounds on the spectral norm of sub-exponential random matrices. *arXiv preprint arXiv:2212.07600*, 2022.

[13] Chandler Davis and William Morton Kahan. The rotation of eigenvectors by a perturbation. iii. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[14] Konstantin Donhauser, Johan Lokna, Amartya Sanyal, March Boedihardjo, Robert Hönig, and Fanny Yang. Sample-efficient private data release for Lipschitz functions under sparsity assumptions. *arXiv preprint arXiv:2302.09680*, 2023.

[15] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 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*, pages 486–503. Springer, 2006.

[16] Cynthia Dwork, Nitin Kohli, and Deirdre Mulligan. Differential privacy in practice: Expose your epsilons! *Journal of Privacy and Confidentiality*, 9(2), 2019.

[17] Cynthia Dwork, Aleksandar Nikolov, and Kunal Talwar. Efficient algorithms for privately releasing marginals via convex relaxations. *Discrete & Computational Geometry*, 53:650–673, 2015.

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

[19] Cynthia Dwork, Kunal Talwar, Abhradeep Thakurta, and Li Zhang. Analyze Gauss: optimal bounds for privacy-preserving principal component analysis. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 11–20, 2014.

[20] Uriel Feige and Eran Ofek. Spectral techniques applied to sparse random graphs. *Random Structures & Algorithms*, 27(2):251–275, 2005.

[21] Moritz Hardt, Katrina Ligett, and Frank McSherry. A simple and practical algorithm for differentially private data release. *Advances in neural information processing systems*, 25, 2012.

[22] Moritz Hardt and Eric Price. The noisy power method: A meta algorithm with applications. *Advances in neural information processing systems*, 27, 2014.

[23] Moritz Hardt and Aaron Roth. Beyond worst-case analysis in private singular vector computation. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 331–340, 2013.

[24] Yiyun He, Roman Vershynin, and Yizhe Zhu. Algorithmically effective differentially private synthetic data. *arXiv preprint arXiv:2302.05552*, 2023.

[25] Hafiz Imtiaz and Anand D Sarwate. Symmetric matrix perturbation for differentially-private principal component analysis. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2339–2343. IEEE, 2016.

[26] Prateek Jain, Chi Jin, Sham M Kakade, Praneeth Netrapalli, and Aaron Sidford. Streaming PCA: Matching matrix Bernstein and near-optimal finite sample guarantees for Oja's algorithm. In *Conference on learning theory*, pages 1147–1164. PMLR, 2016.

[27] Wuxuan Jiang, Cong Xie, and Zhihua Zhang. Wishart mechanism for differentially private principal components analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1), 2016.

[28] Xiaoqian Jiang, Zhanglong Ji, Shuang Wang, Noman Mohammed, Samuel Cheng, and Lucila Ohno-Machado. Differential-private data publishing through component analysis. *Transactions on data privacy*, 6(1):19, 2013.

[29] Michael Kapralov and Kunal Talwar. On differentially private low rank approximation. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*, pages 1395–1414. SIAM, 2013.

[30] Xiaocan Li, Shuo Wang, and Yinghao Cai. Tutorial: Complexity analysis of singular value decomposition and its variants. *arXiv preprint arXiv:1906.12085*, 2019.

[31] Xiyang Liu, Weihao Kong, Prateek Jain, and Sewoong Oh. DP-PCA: Statistically optimal and differentially private PCA. *arXiv preprint arXiv:2205.13709*, 2022.

[32] Xiyang Liu, Weihao Kong, and Sewoong Oh. Differential privacy and robust statistics in high dimensions. In *Conference on Learning Theory*, pages 1167–1246. PMLR, 2022.

[33] Oren Mangoubi and Nisheeth Vishnoi. Re-analyze Gauss: Bounds for private matrix approximation via Dyson Brownian motion. *Advances in Neural Information Processing Systems*, 35:38585–38599, 2022.

[34] Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15:267–273, 1982.

[35] Markus Reiss and Martin Wahl. Nonasymptotic upper bounds for the reconstruction error of PCA. *The Annals of Statistics*, 48(2):1098–1123, 2020.

[36] Justin Thaler, Jonathan Ullman, and Salil Vadhan. 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*, pages 810–821. Springer, 2012.

[37] Jonathan Ullman and Salil Vadhan. 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*, pages 400–416. Springer, 2011.

[38] Giuseppe Vietri, Cedric Archambeau, Sergul Aydore, William Brown, Michael Kearns, Aaron Roth, Ankit Siva, Shuai Tang, and Steven Wu. Private synthetic data for multitask learning and marginal queries. In *Advances in Neural Information Processing Systems*, 2022.

[39] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

[40] Van Vu. Singular vectors under random perturbation. *Random Structures & Algorithms*, 39(4):526–538, 2011.

[41] Ziteng Wang, Chi Jin, Kai Fan, Jiaqi Zhang, Junliang Huang, Yiqiao Zhong, and Liwei Wang. Differentially private data releasing for smooth queries. *The Journal of Machine Learning Research*, 17(1):1779–1820, 2016.

[42] Yi Yu, Tengyao Wang, and Richard J Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

[43] Shuheng Zhou, Katrina Ligett, and Larry Wasserman. Differential privacy with compression. In *2009 IEEE International Symposium on Information Theory*, pages 2718–2722. IEEE, 2009.

Appendix A. Proofs

A.1. **Proof of Theorem 3.1.**

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

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

Now, if there are neighboring datasets $\mathbf{X}$ and $\mathbf{X}'$, suppose $X_k = (X_k^{(1)}, \ldots, X_k^{(d)})^\top$ is a column vector in $\mathbf{X}$ and $X'_k = ({X'_k}^{(1)}, \ldots, {X'_k}^{(d)})^\top$ 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 $\mathbf{X}'$:

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

As the datasets differs on only one data X_k, consider all entry containing X_k in (A.1), we have

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

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. □

A.2. **Proof of Lemma 3.2.**

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

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

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

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

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

$$|\sigma_i - \widehat{\sigma}_i| \le \|\mathbf{A}\|. \qquad (A.3)$$

By von Neumann's trace inequality,

$$\mathrm{tr}(A\widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^{\mathsf{T}}) \le \sum_{i=1}^{d'}\sigma_i(\mathbf{A}). \qquad (A.4)$$

From (A.2), (A.3), and (A.4),

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

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

$$W_2^2(\mu_{\mathbf{X}}, \mu_{\mathbf{Y}}) \le \frac{1}{n}\sum_{i=1}^{n}\|X_i - Y_i\|_2^2 = \frac{1}{n}\|\mathbf{X} - \mathbf{Y}\|_F^2. \qquad (A.5)$$

Therefore (3.2) holds. □

A.3. **Proof of Theorem 3.3.**

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

$$\mathbf{M} = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})(X_i - \overline{X})^{\mathsf{T}} = \sum_{j=1}^{d}\sigma_j v_j v_j^{\mathsf{T}}, \tag{A.6}$$

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

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

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

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

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

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

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

With Lemma 3.2, by definition of the Wasserstein distance, we have

$$\begin{aligned}W_2^2(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^{\mathsf{T}}}, \mu_{\widehat{\mathbf{X}}}) &= \frac{1}{n}\sum_{i=1}^{n}\left\|X_i - \overline{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^{\mathsf{T}}(X_i - \overline{X} - \lambda)\right\|_\infty^2 &\text{(A.7)}\\ &\leq \frac{2}{n}\sum_{i=1}^{n}\left\|X_i - \overline{X} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^{\mathsf{T}}(X_i - \overline{X})\right\|_2^2 + 2\|\lambda\|_2^2 &\text{(A.8)}\\ &= \frac{2}{n}\|\mathbf{Z} - \widehat{\mathbf{V}}_{d'}\widehat{\mathbf{V}}_{d'}^{\mathsf{T}}\mathbf{Z}\|_F^2 + 2\|\lambda\|_2^2 &\text{(A.9)}\\ &\leq 2\sum_{i=d'}^{n}\sigma_i(\mathbf{M}) + 4d'\|\mathbf{A}\| + 2\|\lambda\|_2^2 . &\text{(A.10)}\end{aligned}$$

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

$$\mathbb{E}\|\lambda\|_2^2 = \sum_{j=1}^{d}\mathbb{E}\left|\lambda_j\right|^2 = \frac{2d}{\varepsilon^2 n^2}. \tag{A.11}$$

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 [12, Theorem 1.1]

$$\mathbb{P}\left\{\|\mathbf{A}\| \geq \sigma(C\sqrt{d} + t)\right\} \leq C_0 \exp(-C_1 \min(t^2/4, t/2)). \tag{A.12}$$

And we can further compute the expectation bound for $\|\mathbf{A}\|$ from (A.12) with the choice of $\sigma = \frac{3d^2}{\varepsilon n}$,

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

Combining the two bounds above and (A.10), 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 $x, y \ge 0$,

$$\begin{aligned}
\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) &\le \left(\mathbb{E}\, W_2^2(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}})\right)^{1/2} \\
&\le \sqrt{2\sum_{i>d'} \sigma_i(\mathbf{M})} + \sqrt{4d'\,\mathbb{E}\|\mathbf{A}\|} + \sqrt{2\,\mathbb{E}\|\lambda\|_2^2} \\
&\le \sqrt{2\sum_{i>d'} \sigma_i(\mathbf{M})} + \sqrt{\frac{Cd'd^{2.5}}{\varepsilon n}}.
\end{aligned}$$

□

A.4. **Proof of Proposition 4.2.**

Proof. The privacy guarantee follows from [24, 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 [24, Corollary 4.4], we obtain the accuracy bound. □

A.5. **Proof of Proposition 4.4.**

Proof. The proposition is a direct corollary to the result in [24]. 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, for example, [20, Claim 2.9] and [8, 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 [24, 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. □

A.6. **Proof of Theorem 5.1.**

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

(1) $\mathcal{M}_1(\mathbf{X}) = \widehat{\mathbf{M}}$ is to compute the private sample covariance matrix with Algorithm 2.
(2) $\mathcal{M}_2(\mathbf{X}) = \overline{X} + \lambda$ is to compute the private sample mean.
(3) $\mathcal{M}_3(\mathbf{X}, y, \Sigma)$ 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 $\mathcal{M}_2$ and $\mathcal{M}_1$, respectively). This step outputs synthetic data $\mathbf{X}' = (X_1', \dots, X_m')$ on a linear subspace.
(4) $\mathcal{M}_4(\mathbf{X}, \mathbf{X}')$ is to shift the dataset to $\{X_i' + \overline{X} + \lambda'\}_{i=1}^m$.
(5) Metric projection.

It suffices to show that the data before metric projection has already been differentially private. We will need to apply Lemma 2.1 several times.

With respect to the input $\mathbf{X}$ while fixing other input variables, we know that $\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3, \mathcal{M}_4$ are all $\varepsilon/4$-differentially private. Therefore, by using Lemma 2.1 iteratively, the composition algorithm

$$\mathcal{M}_4(\mathbf{X}, \mathcal{M}_3(\mathbf{X}, \mathcal{M}_2(\mathbf{X}), \mathcal{M}_1(\mathbf{X})))$$

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

A.7. **Proof of Theorem 5.2.**

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) $\mathcal{M}(\mathbf{X})$ is the data after metric projection, which is the output of the whole algorithm.

For the metric projection step, by Proposition 4.5, we have that

$$\begin{aligned} W_1(\mu_{\mathbf{X}}, \mu_{\mathcal{M}(\mathbf{X})}) &\leq W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}) + W_1(\mu_{\mathbf{X}''}, \mu_{\mathcal{M}(\mathbf{X})}) && \text{(A.13)} \\ &\leq 2W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}). && \text{(A.14)} \end{aligned}$$

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

$$\begin{aligned} W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}''}) &= W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\mathbf{X}'+\lambda'\mathbf{1}^\mathsf{T}}) && \text{(A.15)} \\ &\leq W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) + W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) + W_1(\mu_{\mathbf{X}'}, \mu_{\mathbf{X}'+\lambda'}) && \text{(A.16)} \\ &\leq W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) + W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) + \left\|\lambda'\right\|_\infty. && \text{(A.17)} \end{aligned}$$

Note that $W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}})$ is the projection error we bound in Theorem 3.3, and $W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'})$ is treated in the accuracy analysis of subroutines in Section 4. Moreover, we have

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

Here in the last step we use $\mathbb{E}\|\lambda\|_2 \leq \frac{C\sqrt{d}}{\varepsilon n}$ in (A.11). Since λ' is a sub-exponential random vector, the following bound also holds for some absolute constant $C > 0$:

$$\mathbb{E}\left\|\lambda'\right\|_\infty \leq \frac{C \log d}{\varepsilon n}. \tag{A.18}$$

Hence

$$\begin{aligned}
&\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathcal{M}(\mathbf{X})}) && \text{(A.19)}\\
&\le 2\,\mathbb{E}\, W_1(\mu_{\mathbf{X}}, \mu_{\mathbf{X}'+(\overline{X}+\lambda')\mathbf{1}^\mathsf{T}}) && \text{(A.20)}\\
&\le 2\,\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) + 2\,\mathbb{E}\, W_1(\mu_{\widehat{\mathbf{X}}}, \mu_{\mathbf{X}'}) + 2\,\mathbb{E}\big\|\lambda'\big\|_\infty && \text{(A.21)}\\
&\le 2\sqrt{2\sum_{i>d'}\sigma_i(\mathbf{M})} + 2\sqrt{\frac{Cd'd^{2.5}}{\varepsilon n}} + 2C\sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'} + \frac{2C\log d}{\varepsilon n} && \text{(A.22)}\\
&\lesssim \sqrt{\sum_{i>d'}\sigma_i(\mathbf{M})} + \sqrt{\frac{d}{d'}}(\varepsilon n)^{-1/d'} + \sqrt{\frac{d'd^{2.5}}{\varepsilon n}}, && \text{(A.23)}
\end{aligned}$$

where the first inequality is from (A.14), the second inequality is from (A.17), and the third inequality is due to Theorem 3.3, Proposition 4.2, and Proposition 4.4. □

A.8. Proof of Proposition 4.5.

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 $X_1, X_2, \dots, X_n$, 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

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

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.

The expectation bound follows from (A.17) and (A.22). □

A.9. **Results when $d' = 1$ with extra assumptions.** We start with the following lemma based on the Davis-Kahan theorem [42].

Lemma A.1. *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 $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_d$ are the singular values of $\mathbf{M}$ and $v_1, \dots, v_d$ are corresponding orthonormal eigenvectors. Let $\widehat{\mathbf{M}} = \frac{1}{n}\mathbf{X}\mathbf{X}^\mathsf{T} + \mathbf{A}$ with orthonormal eigenvectors $\widehat{v}_1, \dots, \widehat{v}_d$, 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

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

Proof. We have that

$$\begin{aligned}
\frac{1}{n}\|\mathbf{X}-\widehat{v}_1\widehat{v}_1^{\mathsf{T}}\mathbf{X}\|_F^2 &= \frac{1}{n}\|\mathbf{X}-v_1v_1^{\mathsf{T}}\mathbf{X}+v_1v_1^{\mathsf{T}}\mathbf{X}-\widehat{v}_1\widehat{v}_1^{\mathsf{T}}\mathbf{X}\|_F^2 \\
&\le \frac{2}{n}\left(\|\mathbf{X}-v_1v_1^{\mathsf{T}}\mathbf{X}\|_F^2+\|v_1v_1^{\mathsf{T}}\mathbf{X}-\widehat{v}_1\widehat{v}_1^{\mathsf{T}}\mathbf{X}\|_F^2\right) \\
&= 2\sum_{i>d'}\sigma_i+\frac{2}{n}\left\|\left(v_1v_1^{\mathsf{T}}-\widehat{v}_1\widehat{v}_1^{\mathsf{T}}\right)\mathbf{X}\right\|_F^2 \\
&\le 2\sum_{i>d'}\sigma_i+\frac{2}{n}\left\|v_1v_1^{\mathsf{T}}-\widehat{v}_1\widehat{v}_1^{\mathsf{T}}\right\|^2\|\mathbf{X}\|_F^2 . \qquad \text{(A.24)}
\end{aligned}$$

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 [42, Corollary 1], we have

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

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

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

□

Compared to Lemma 3.2, 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 [35, Proposition 2.2]. With Lemma A.1, we are able to improve the accuracy rate for the noisy projection step as follows.

Theorem A.2. *When $d'=1$, assume that $\sigma_1(\mathbf{M})=\|\mathbf{M}\|>0$. For the output $\widehat{\mathbf{X}}$ in Algorithm 3, we have*

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

where $\sigma_1\ge\cdots\ge\sigma_d\ge 0$ are singular values of $\mathbf{M}$.

Proof. Similar to the proof of Theorem 3.3, we can define $Z_i=X_i-\overline{X}$ and deduce that

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

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 $\sqrt{x+y}\le\sqrt{x}+\sqrt{y}$ for $x,y\ge 0$,

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

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

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

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

As any projection map has spectral norm 1, we have $\left\| v_1 v_1^\mathsf{T} - \widehat{v}_1\widehat{v}_1^\mathsf{T} \right\| \le 2$. Applying (A.24), we have

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

Hence

$$\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) \lesssim \sqrt{\sum_{i>1}\sigma_i} + \mathbb{E}\|\lambda\|_2 \lesssim \sqrt{\sum_{i>1}\sigma_i} + \frac{\sqrt{d}}{\varepsilon n}. \tag{A.25}$$

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

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

For any fixed δ, by Lemma A.1,

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

So we have the Wasserstein distance bound

$$\begin{aligned}\mathbb{E}\, W_1(\mu_{\mathbf{X}-\overline{X}\mathbf{1}^\mathsf{T}}, \mu_{\widehat{\mathbf{X}}}) &\le \sqrt{2\sum_{i>1}\sigma_i} + \sqrt{\frac{32d}{\sigma_1}}\,\mathbb{E}\|\mathbf{A}\| + \sqrt{2}\,\mathbb{E}\|\lambda\|_2 && \text{(A.26)} \\ &\le \sqrt{2\sum_{i>1}\sigma_i} + \sqrt{\frac{32d}{\sigma_1}}\frac{d^{2.5}}{\varepsilon n} + \frac{\sqrt{2d}}{\varepsilon n} && \text{(A.27)} \\ &\le \sqrt{2\sum_{i>1}\sigma_i} + \frac{Cd^3}{\sqrt{\sigma_1}\varepsilon n}. && \text{(A.28)}\end{aligned}$$

From (A.6),

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

Combining the two cases (A.25) and (A.28), we deduce the result. □

Proof of Theorem 6.1. Following the steps in the proof of Theorem 3.3, we obtain

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

where for the second inequality, we apply the bound from [24, Theorem 1.1] for the second term, and we use (A.18) for the third term. □

Department of Mathematics, University of California Irvine
Email address: yiyunh4@uci.edu

Department of Mathematics and Center of Data Science and Artificial Intelligence Research, University of California Davis
Email address: strohmer@math.ucdavis.edu

Department of Mathematics, University of California Irvine
Email address: rvershyn@uci.edu

Department of Mathematics, University of California Irvine
Email address: yizhe.zhu@uci.edu