Hindawi
Security and Communication Networks
Volume 2022, Article ID 5612794, 13 pages
https://doi.org/10.1155/2022/5612794

Research Article

Multiparty Data Publishing via Blockchain and Differential Privacy

Zhen Gu,[1] **Kejia Zhang**,[1,2] **and Guoyin Zhang**[1]

[1]*College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China*
[2]*School of Mathematical Science, Heilongjiang University, Harbin 150080, China*

Correspondence should be addressed to Kejia Zhang; zhangkejia@hlju.edu.cn

Received 21 January 2022; Accepted 30 March 2022; Published 9 May 2022

Academic Editor: Yuling Chen

Data are distributed between different parties. Collecting data from multiple parties for analysis and mining will serve people better. However, it also brings unprecedented privacy threats to the participants. Therefore, safe and reliable data publishing among multiple data owners is an urgent problem to be solved. We mainly study the problem of privacy protection in data publishing. For a centralized scenario, we propose the LDA-DP algorithm. First, the within-class mean vectors and the pooled within-class scatter matrix are perturbed by the Gaussian noise. Second, the optimal projection direction vector with differential privacy is obtained by the Fisher criterion. Finally, the low-dimensional projection data of the original data are obtained. For distributed scenarios, we propose the Mul-LDA-DP algorithm based on a blockchain and differential privacy technology. First, the within-class mean vectors and within-class scatter matrices of local data are perturbed by the Gaussian noise and uploaded to the blockchain network. Second, the projection direction vector is calculated in the blockchain network and returned to the data owner. Finally, the data owner uses the projection direction vector to generate low-dimensional projection data of the original data and upload it to the blockchain network for publishing. Furthermore, in a distributed scenario, we propose a correlated noise generation scheme that uses the additivity of the Gaussian distribution to mitigate the effects of noise and can achieve the same noise level as the centralized scenario. We measure the utility of the published data by the SVM misclassification rate. We conduct comparative experiments with similar algorithms on different real data sets. The experimental results show that the data released by the two algorithms can maintain good utility in SVM classification.

1. Introduction

With the development of science and technology, effective data collection and analysis can help people make better decisions in production. For example, analyzing the information of the patient can help doctors improve the accuracy of diagnosis and level of medical services, and analyzing the trajectory data can improve city traffic congestion. The data contain sensitive information and need to be processed for privacy protection before publishing [1, 2]. There have been some studies on privacy preserving data publishing. For example, the k -anonymity privacy protection technology [3], the encryption technology [4, 5], the blockchain technology [6–8], and differential privacy technology [9–11]. Differential privacy has been widely used for privacy protection in recent years, the principle of differential privacy is to add random noise to data, which makes the attacker unable to distinguish the original input data. Differential privacy can quantitatively measure the degree of privacy protection and can resist attacks from attackers with background knowledge. Privacy preserving data publishing based on differential privacy has become a research hot spot [12–15].

However, in the distributed scenario, data are possessed by multiple data owners. Data from a single data owner may not be sufficient for statistical learning, and aggregating data by a single data owner may not be possible. For example [16], in Table 1, the data are possessed by three data owners. Each row in Table 1 represents the information of an individual, where records 1 to 4 are from data owner 1, records 5 to 8 are

Table 1: Aggregated dataset of each data owner.

ID	Age	Job	Gender	hours-per-week	income
1	39	Shopkeeper	Male	40	> 50K
2	55	Lawyer	Male	13	≤ 50K
3	38	Dancer	Male	20	≤ 50K
4	30	Dancer	Male	25	≤ 50K
5	28	Builder	Female	40	> 50K
6	37	Dancer	Female	23	≤ 50K
7	49	Teacher	Female	16	≤ 50K
8	52	Builder	Male	45	> 50K
9	31	Lawyer	Female	50	> 50K
10	42	Builder	Male	40	> 50K

from data owner 2, and records 9 to 10 are from data owner 3. Simply integrating and publishing the data from each data owner will cause a serious privacy leakage. Sharing and exchange of data in a distributed environment requires security guarantees. In order to solve the proposed problem, we make the following contributions:

(1) We propose two algorithms which are called LDA-DP and Mul-LDA-DP. The LDA-DP algorithm is used for privacy protection of data publishing in centralized scenario, and the Mul-LDA-DP algorithm is used for privacy protection of data publishing in distributed scenario.

(2) In the distributed scenario, the data owners cooperate with each other to publish a projection data set which satisfies differential privacy. In order to improve the utility of the published data in the distributed scenario, we propose a correlated noise generation scheme that uses the additivity of the Gaussian distribution to mitigate the effects of noise and can achieve the same noise level as the centralized scenario.

(3) We conduct experiments on different data sets. The experimental results show that the data released by LDA-DP and Mul-LDA-DP algorithms can maintain good utility in SVM classification.

2. Related Work

In this section, we introduce the research status of privacy preserving data publishing in centralized scenario and distributed scenario, respectively.

2.1. Privacy Preserving Data Publishing in Centralized Scenario. Blum et al. [17] proposed the sublinear query (SULQ) input perturbation framework which adds noise to the covariance matrix, the framework can only be used for querying the projected subspace. Chaudhuri et al. [18] proposed the PPCA algorithm which is the improvement of SUQL algorithm. The PPCA algorithm randomly samples a k-dimensional subspace which ensures differential privacy and is biased toward high utility. Both SUQL and PPCA procedures are differentially private approximations to the top-k subspace. Zhang et al. [19] proposed the PrivBayes algorithm; first, they constructed a Bayesian network with differential privacy, and then they used the Bayesian network to generate a data set for publication. Chen et al. [20] presented the JTree algorithm. First, they explored the relationship between the attributes based on the sparse vector sampling technology, and then they constructed a Markov network that satisfies differential privacy and generated a synthetic data set for publication. Zhang et al. [21] proposed the PrivHD algorithm based on the JTree. They used high-pass filtering techniques to speed up the construction of Markov network and built a better joint tree for generating synthetic data set for publication. Xu et al. [22] proposed the DPPro algorithm; first, they randomly projected the original high-dimensional data into a low-dimensional space, and then they added noise to the projection vector and low-dimensional projection data; finally, they released the low-dimensional projection data. Zhang et al. [23] presented the PrivMN method. They constructed a Markov model with differential privacy, and then used the Markov model to generate a synthetic data set for publication. The algorithms mentioned above are mainly used for privacy preserving data publishing in centralized scenarios.

2.2. Privacy Preserving Data Publishing in Distributed Scenario. There are fewer researches on privacy protection of horizontally partitioned data publication. Ge et al. [24] proposed a distributed principal component analysis (DPS-PCA) algorithm with differential privacy; first, data owners collaborated to analyze the principal components, while protecting the private information, and then they released low-dimensional subspaces of high-dimensional sparse data. Wang et al. [25] proposed an efficient and scalable protocol for computing principal components in a distributed environment. First, the data owner encrypted the shared data and sent them to the semitrusted third party, then the semitrusted third party performed a private aggregation algorithm on the encrypted data and sent the aggregated data to data user for calculating the principal components. Imtiaz et al. [26] presented a distributed principal component analysis (DPdisPCA) algorithm with differential privacy. Each data owner used Gaussian noise to perturbed the local covariance matrix, and with the assistance of a semitrusted third party to calculate the principal components while ensuring local data privacy. Alhadidi et al. [27] proposed a two-party data publishing algorithm with differential privacy. They first presented a two-party protocol for the exponential mechanism which can be used as a subprotocol, the data released by this algorithm are suitable for classification tasks. Cheng et al. [28] proposed a differential privacy sequential update of the Bayesian network algorithm which is called DP-SUBN^3, data owners collaboratively constructed the Bayesian network, data owners can treat the intermediate results as prior knowledge to construct the Bayesian network, and then they used the Bayesian network to generate a data set for publication. Wang et al. [29] proposed a distributed differential privacy anonymous algorithm and guaranteed that each step of the algorithm satisfies the definition of secure two-party computation. This is the first research about differentially private data publishing for arbitrarily partitioned data. In our prior work [16], we proposed the PPCA-DP-MH algorithm. First, data owners and a semitrusted third party cooperated to reduce the dimension of high-dimensional data to obtain the top k principal components that satisfy differential privacy, and then each data owner used the generative model of probabilistic principal component analysis to generate a data set with the same scale as the original data for publication. Different from the prior work [16], this paper uses the linear discriminant analysis to publish the projection data with differential privacy. Linear discriminant analysis can retain the class information of the data while reducing the dimension, which is beneficial to maintain the utility of the published data in classification.

3. Preliminaries

3.1. Linear Discriminant Analysis (LDA). Linear discriminant analysis proposed by Fisher is one of the most widely used and extremely effective methods in the field of dimensionality reduction and pattern recognition. Its typical applications include face recognition, target tracking and detection, credit card fraud detection, and speech recognition. The idea of linear discriminant analysis for binary classification is to choose the projection direction so that the samples of different classes after projection are as far apart as possible and the samples within each class are as clustered as possible. We denote the data set as $X = X^{(1)} \bigcup X^{(2)}$, $X^{(k)} = \left\{\mathbf{x}_1^{(k)}, \mathbf{x}_2^{(k)}, \ldots, \mathbf{x}_{N^{(k)}}^{(k)}\right\}$, $k = 1, 2$. $N = N^{(1)} + N^{(2)}$. The within-class mean vector of samples in the original sample space is as follows:

$$\mu^{(k)} = \frac{1}{N^{(k)}} \sum_{\mathbf{x}\in X^{(k)}} \mathbf{x}, \quad k = 1, 2. \tag{1}$$

The between-class scatter matrix is as follows:

$$S_b = \left(\mu^{(1)} - \mu^{(2)}\right)\left(\mu^{(1)} - \mu^{(2)}\right)^{\mathrm{T}}. \tag{2}$$

The within-class scatter matrix is as follows:

$$S^{(k)} = \sum_{\mathbf{x}\in X^{(k)}} \mathbf{x}\mathbf{x}^{\mathrm{T}} - N^{(k)}\mu^{(k)}\left(\mu^{(k)}\right)^{\mathrm{T}}, k = 1, 2. \tag{3}$$

Then, the pooled within-class scatter matrix is as follows:

$$S_w = S^{(1)} + S^{(2)}. \tag{4}$$

It can also be expressed as follows:

$$S_w = \sum_{k=1}^{2} \sum_{\mathbf{x}\in X^{(k)}} \mathbf{x}\mathbf{x}^{\mathrm{T}} - \sum_{k=1}^{2} N^{(k)}\mu^{(k)}\left(\mu^{(k)}\right)^{\mathrm{T}}. \tag{5}$$

The criterion of Fisher is as follows:

$$\max_{\mathbf{w}} \frac{\mathbf{w}^{\mathrm{T}} S_b^{-1} \mathbf{w}}{\mathbf{w}^{\mathrm{T}} S_w^{-1} \mathbf{w}}. \tag{6}$$

Using the Lagrange multiplier method to find the optimal projection direction vector, we obtain the following:

$$\mathbf{w} = S_w^{-1}\left(\mu^{(1)} - \mu^{(2)}\right). \tag{7}$$

The result of linear discriminant analysis only gives the optimal projection direction, and does not give a clear classification result.

3.2. Differential Privacy. Differential privacy provides a rigorous privacy protection for sensitive information, it can be quantified by mathematical formulas. The essence of differential privacy is to use noise to randomly perturb the output results, so that it is difficult to distinguish the original input data according to the output results.

Definition 1. [30] A randomized algorithm $\mathscr{M}$ is ε -indistinguishable if for any two neighboring databases D and $\widehat{D}$ differing in a single entry, and for all O⊆Range$(\mathscr{M})$:

$$\left|\ln \frac{P_r[\mathscr{M}(D) \in O]}{P_r[\mathscr{M}(\widehat{D}) \in O]}\right| \le \varepsilon, \tag{8}$$

where ε is a small positive real number.

When ε is small, $\ln(1 + \varepsilon) \approx \varepsilon$, so $P_r[\mathscr{M}(D) \in O]/P_r[\mathscr{M}(\widehat{D}) \in O] \in [1 - \varepsilon, 1 + \varepsilon]$, ε is used to control the probability ratio of algorithm $\mathscr{M}$ to obtain the same output on two neighboring databases, which reflects the level of privacy protection that $\mathscr{M}$ can provide.

Definition 2 [30]. A randomized algorithm $\mathscr{M}$ is (ε, δ) differential privacy, if for any two neighboring databases D and $\widehat{D}$ differing in a single entry, and for any O(O⊆Range$(\mathscr{M})$) there is the following:

$$P_r\{\mathscr{M}(D) \in O\} \le e^{\varepsilon} P_r\left\{\mathscr{M}(\widehat{D}) \in O\right\} + \delta, \tag{9}$$

where ε is a small positive real number called privacy budget and δ is a small positive real number. It is also called δ-approximate ε-indistinguishability.

Definition 3. is the relaxed version of differential privacy. When $\delta = 0$, it becomes Definition 1, which is the strict version of differential privacy. Formula (9) means that it is allowed to break the limit of formula (8) with a small probability δ.

Theorem 1 ([31]). *The sufficient condition for the random function $\mathscr{M}$ to satisfy (ε, δ) differential privacy is as follows:*

$$P_r\left\{\left|\ln \frac{P_r[\mathscr{M}(D) \in O]}{P_r[\mathscr{M}(\widehat{D}) \in O]}\right| > \varepsilon\right\} \le \delta, \quad \text{O⊆Range}(\mathscr{M}). \tag{10}$$

Theorem 2 (Sequential Composition) [31]. *Let $\mathscr{M}_i$ be an $(\varepsilon_i, \delta_i)$ differentially private algorithm, $i = 1, 2, \ldots, n$, then for the same data set D, the combined algorithm $\mathscr{M}(\mathscr{M}_1(D), \mathscr{M}_2(D), \ldots, \mathscr{M}_n(D))$ is $(\sum_{i=1}^{n} \varepsilon_i, \sum_{i=1}^{n} \delta_i)$ differential privacy.*

Theorem 3 (Parallel Composition) [31]. *Let $\mathscr{M}_i$ be an $(\varepsilon_i, \delta_i)$ differentially private algorithm, $i = 1, 2, \ldots, n$, $D_1, D_2, \ldots, D_n$ are disjoint data sets, the combined algorithm $\mathscr{M}(\mathscr{M}_1(D_1), \mathscr{M}_2(D_2), \ldots, \mathscr{M}_n(D_n))$ is $\max_{1\le i\le n}(\varepsilon_i, \delta_i)$ differential privacy.*

Theorem 4 (Post Processing) [31]. *Let $\mathscr{M}\colon D \longrightarrow R$ be a randomized algorithm that is (ε, δ) differential privacy, let $f\colon R \longrightarrow R'$ be an arbitrary mapping, then $f^{\circ}\mathscr{M}\colon D \longrightarrow R'$ is (ε, δ) differential privacy.*

4. Proposed Methods

In this section, we will propose two algorithms which are called LDA-DP and Mul-LDA-DP. The LDA-DP algorithm is used for privacy protection of data publishing in the centralized scenario, and the Mul-LDA-DP algorithm is used for privacy protection of data publishing in the distributed scenario. Without loss of generality, we assume that all individual data in this paper are normalized to p -dimensional unit vectors.

4.1. LDA-DP Algorithm. In this section, we propose the LDA-DP algorithm for centralized data publishing.

4.1.1. Problem Statement and Algorithm Proposed. The data set X contains two classes of data individuals denoted as $X = X^{(1)} \bigcup X^{(2)}$, where $X^{(k)} = \left\{\mathbf{x}_1^{(k)}, \mathbf{x}_2^{(k)}, \ldots, \mathbf{x}_{N^{(k)}}^{(k)}\right\}$, $k = 1, 2$. Our goal is to protect the privacy information of the original data from being leaked while publishing the projection data of the original data.

In order to solve this problem, we propose the LDA-DP algorithm, which is mainly divided into two stages. First, we use the Gaussian mechanism of differential privacy to perturb the within-class mean vectors $\mu^{(k)}\,(k = 1, 2)$. Second, we use the Gaussian mechanism to perturb the pooled within-class scatter matrix S_w. Finally, we get the projection direction vector $\mathbf{w}$ that satisfies (ε, δ) differential privacy and publish the low-dimensional projected data of the original data. The specific details are in Algorithm 1.

4.1.2. Privacy Analysis of LDA-DP Algorithm

Theorem 5. *The within-class mean vector $\mu^{(k)}\,(k = 1, 2)$ in Algorithm 1 satisfies $(\varepsilon_1, \delta_1)$ differential privacy when each entry of $\mathbf{g}^{(k)}\,(k = 1, 2)$ is sampled from $N(0, \sigma_1^2)$, where $\sigma_1 \geq p^{3/2}\sqrt{ln2/\pi\delta_1^2} + \sqrt{p^3 ln2/\pi\delta_1^2 + 2\varepsilon_1}/\varepsilon_1$, $\quad 0 < \delta_1 < \sqrt{2/\pi}$.*

Proof. We denote the two neighboring data sets are $X = X^{(1)} \bigcup X^{(2)}$ and $\widehat{X} = \widehat{X}^{(1)} \bigcup \widehat{X}^{(2)}$, where only one individual is different, without losing general assumption. Suppose the different individuals are in $X^{(1)}$ and $\widehat{X}^{(1)}$, we denote them as $\mathbf{x}_{N^{(1)}} \neq \widehat{\mathbf{x}}_{N^{(1)}}$, they are p -dimensional unit vector. We denote $\mathbf{a} = \sum_{\mathbf{x} \in X^{(1)}} \mathbf{x}$ and $\widehat{\mathbf{a}} = \sum_{\mathbf{x} \in X^{(1)}} \mathbf{x}$, let $\mathbf{c} = \mathbf{a} + \mathbf{g}^{(1)}$ and $\widehat{\mathbf{c}} = \widehat{\mathbf{a}} + \widehat{\mathbf{g}}^{(1)}$, each entry of $\mathbf{g}^{(1)}$ and $\widehat{\mathbf{g}}^{(1)}$ is sampled from $N(0, \sigma_1^2)$.

The log ratio of the probabilities $\mathbf{c}$ and $\widehat{\mathbf{c}}$ at a point $\mathbf{h}$ is $|\ln((P\{\mathbf{c} = \mathbf{h}|X\})/(P\{\widehat{\mathbf{c}} = \mathbf{h}|\widehat{X}\}))|$, the numerator in the ratio describes the probability of seeing $\mathbf{h}$ when the data set is X, the denominator corresponds the probability of seeing this same value when the data set is $\widehat{X}$.

By Theorem 1, we will to find the value of σ_1 such that the inequality $|\ln((P\{\mathbf{c} = \mathbf{h}|X\})/(P\{\widehat{\mathbf{c}} = \mathbf{h}|\widehat{X}\}))| = |\ln((P\{\mathbf{h} - \mathbf{a}|X\})/(P\{\mathbf{h} - \widehat{\mathbf{a}}|\widehat{X}\}))| \leq \varepsilon_1$ holds at least with probability $1 - \delta_1$.

$$\begin{aligned}\left|\ln\frac{P\{\mathbf{h} - \mathbf{a}|X\}}{P\{\mathbf{h} - \widehat{\mathbf{a}}|\widehat{X}\}}\right| &= \frac{1}{2\sigma_1^2}\left|\sum_{i=1}^{p}\left[(h_i - \widehat{a}_i)^2 - (h_i - a_i)^2\right]\right| \\ &= \frac{1}{2\sigma_1^2}\sum_{i=1}^{p}\left|\left[2(h_i - a_i)(a_i - \widehat{a}_i) + (a_i - \widehat{a}_i)^2\right]\right| \\ &\leq \frac{1}{2\sigma_1^2}\sum_{i=1}^{p} 2\left|(h_i - a_i)(x_{N^{(1)}i} - \widehat{x}_{N^{(1)}i})\right| + \frac{1}{2\sigma_1^2}\sum_{i=1}^{p}(x_{N^{(1)}i} - \widehat{x}_{N^{(1)}i})^2.\end{aligned} \tag{11}$$

Using the Lagrange multiplier method, we can get the maximum value of the objective function $\sum_{i=1}^{p}(|x_{N^{(1)}i}| + |\widehat{x}_{N^{(1)}i}|)$ is $2\sqrt{p}$ under the condition of $\sum_{i=1}^{p}(x_{N^{(1)}i})^2 = 1, \sum_{i=1}^{p}(\widehat{x}_{N^{(1)}i})^2 = 1$.

Then, we can obtain: $\sum_{i=1}^{p}|(x_{N^{(1)}i}) - (\widehat{x}_{N^{(1)}i})| \leq \sum_{i=1}^{p}(|x_{N^{(1)}i}| + |\widehat{x}_{N^{(1)}i}|) \leq 2\sqrt{p}$. Similarly, we can obtain the following:

$$\sum_{i=1}^{p}(x_{N^{(1)}i}) - (\widehat{x}_{N^{(1)}i})^2 \leq 4. \tag{12}$$

So, $1/2\sigma_1^2\sum_{i=1}^{p} 2|(h_i - a_i)(x_{N^{(1)}i}) - (\widehat{x}_{N^{(1)}i})| \leq 2p^{3/2}r/\sigma_1^2$, where $|g_i^{(1)}| = |h_i - a_i| \leq r$, for all i, and $1/2\sigma_1^2\sum_{i=1}^{p}(x_{N^{(1)}i} - \widehat{x}_{N^{(1)}i})^2 \leq 2/\sigma_1^2$.

Then, $|\ln((P\{\mathbf{b} = \mathbf{h}|X\})/(P\{\widehat{\mathbf{b}} = \mathbf{h}|\widehat{X}\}))| \leq (2p^{3/2}r + 2)/\sigma_1^2$, this quantity is bounded by ε_1 whenever $|g_i^{(1)}| \leq r \leq (\varepsilon_1\sigma_1^2 - 2)/2p^{3/2}$.

To ensure privacy loss bounded by ε_1 with probability at least $1 - \delta_1$, we require to find σ_1 that satisfies this inequality $P_r\left\{|g_i^{(1)}| \geq \varepsilon_1\sigma_1^2 - 2/2p^{3/2}\right\} \leq \delta_1$, due to symmetry, we will find σ_1 such that $P_r\left\{g_i^{(1)} \geq (\varepsilon_1\sigma_1^2 - 2)/2p^{3/2}\right\} \leq \delta_1/2$.

```
Input: Data sets $X$, privacy parameters $(\varepsilon_1, \delta_1)$, $(\varepsilon_2, \delta_2)$
Output: Projection direction vector $\mathbf{w}$, projection data $\widetilde{X}$
for $k = 1$ to $2$ do
    Set $\sigma_1 = p^{3/2}\sqrt{\ln 2/\pi\delta_1^2} + \sqrt{p^3\ln 2/\pi\delta_1^2 + 2\varepsilon_1}/\varepsilon_1$, which generates a $p$ dimension noise vector $\mathbf{g}^{(k)}$; each entry is sampled from $N(0, \sigma_1^2)$
    Computes $\mu^{(k)} = 1/N^{(k)}(\sum_{\mathbf{x}\in X^{(k)}}\mathbf{x} + \mathbf{g}^{(k)})$
end for
return $\mu^{(k)}$, $k = 1, 2$
Set $\sigma_2 = (p+1)\sqrt{\ln 2/\pi\delta_2^2} + \sqrt{(p+1)^2\ln 2/\pi\delta_2^2 + 4\varepsilon_2}/2\varepsilon_2$, which generates a $p \times p$ random matrix $G$. Let $G$ be a symmetric matrix with the upper triangle (including the diagonal) entries are sampled from $N(0, \sigma_2^2)$ and make the symmetrical position entries in the lower triangle matrix equal to the upper triangle.
Computes $S_w = \sum_{\mathbf{x}\in X^{(k)}}^{2}\mathbf{x}\mathbf{x}^{\mathrm{T}} - \sum_{k=1}^{2} N^{(k)}\mu^{(k)}(\mu^{(k)})^{\mathrm{T}} + G$
Computes $\mathbf{w} = S_w^{-1}(\mu^{(1)} - \mu^{(2)})$
Computes $\widetilde{X} = X\mathbf{w}$
```

Algorithm 1: LDA-DP algorithm.

The tail bound is as follows:

$$P_r\{g_i^{(1)} > t\} = \int_t^{+\infty}\frac{1}{\sqrt{2\pi}\sigma_1}e^{-\frac{x^2}{2\sigma_1^2}}\mathrm{d}x \overset{x=t+y}{=} \int_0^{+\infty}\frac{1}{\sqrt{2\pi}\sigma_1}e^{-\frac{(t+y)^2}{2\sigma_1^2}}\mathrm{d}y,$$
$$\leq \frac{1}{\sqrt{2\pi}\sigma_1}e^{-\frac{t^2}{2\sigma_1^2}}\int_0^{+\infty}e^{-\frac{ty}{\sigma_1^2}}\mathrm{d}y \leq \frac{1}{t}\frac{\sigma_1}{\sqrt{2\pi}}e^{-\frac{t^2}{2\sigma_1^2}}. \tag{13}$$

We let $t = (\varepsilon_1\sigma_1^2 - 2)/2p^{3/2}$, then $1/t\sigma_1/\sqrt{2\pi}e^{-t^2/2\sigma_1^2} \leq \delta_1/2$, then we obtain the following:

$$\ln\frac{t}{\sigma_1} + \frac{t^2}{2\sigma_1^2} \geq \ln\frac{2}{\sqrt{2\pi}\delta_1}. \tag{14}$$

When $\sigma_1 \geq [2p^{3/2} + (\sqrt{4p^3 + 8\varepsilon_1})]/2\varepsilon_1$, the first term in (14) is non-negative. To make the inequality (14) hold, we let $t^2/2\sigma_1^2 = 1/2\sigma_1^2(\varepsilon_1\sigma_1^2 - 2/2p^{3/2})^2 \geq \ln 2/\sqrt{2\pi}\delta_1$, then we obtain the following:

$$\sigma_1 \geq \frac{p^{(3/2)}\sqrt{\ln\left(2/\pi\delta_1^2\right)} + \sqrt{p^3\ln\left(2/\pi\delta_1^2\right) + 2\varepsilon_1}}{\varepsilon_1}, \quad 0 < \delta_1 < \sqrt{\frac{2}{\pi}}. \tag{15}$$

Theorem 6. *The pooled within-class scatter matrix S_w in Algorithm 1 satisfies $(\varepsilon_2, \delta_2)$ differential privacy, when each entry in the symmetric random matrix G is sampled from $N(0, \sigma_2^2)$, where*

$$\sigma_2 \geq (p+1)\sqrt{\ln\frac{2}{\pi\delta_2^2}} + \sqrt{(p+1)^2\ln\frac{2}{\pi\delta_2^2} + 4\varepsilon_2/2\varepsilon_2},$$
$$0 < \delta_2 < \sqrt{\frac{2}{\pi}}. \tag{16}$$

Proof. Two neighboring data sets are $X = X^{(1)}\bigcup X^{(2)}$ and $\widehat{X} = \widehat{X}^{(1)}\bigcup\widehat{X}^{(2)}$, where only one entry is different, without losing general assumption, suppose the different entry are in $X^{(1)}$ and $\widehat{X}^{(1)}$ and denoted them as $\mathbf{x}_{N^{(1)}} \neq \widehat{\mathbf{x}}_{N^{(1)}}$.

Because $\mu^{(k)}$ $(k = 1, 2)$ in (5) satisfies differential privacy has been proved by Theorem 5 which can be treated as a constant in (5), so if we want to prove that this theorem holds, it is only necessary to prove that the first item $\sum_{k=1}^{2}\sum_{\mathbf{x}\in X^{(k)}}\mathbf{x}\mathbf{x}^{\mathrm{T}}$ in (5) satisfies $(\varepsilon_2, \delta_2)$ differential privacy after adding random matrix G.

We denote $B = \sum_{k=1}^{2}\sum_{\mathbf{x}\in X^{(k)}}\mathbf{x}\mathbf{x}^{\mathrm{T}}$, $\widehat{B} = \sum_{k=1}^{2}\sum_{\mathbf{x}\in X^{(k)}}\mathbf{x}\mathbf{x}^{\mathrm{T}}$, let $C = B + G$ and $\widehat{C} = \widehat{B} + \widehat{G}$, G and $\widehat{G}$ are two independent symmetric random matrices with the upper triangle (including the diagonal) entries are sampled from $N(0, \sigma_2^2)$, and make the symmetrical position entries in the lower triangle matrix equal to the upper triangle.

The log ratio of the probabilities C and $\widehat{C}$ at a point H is $|\ln((P\{C = H|X\})/(P\{\widehat{C} = H|\widehat{X}\}))|$.

By Theorem 1, we need to find the value of σ_2 such that the inequality $|\ln((P\{C = H|X\})/(P\{\widehat{C} = H|\widehat{X}\}))| \leq \varepsilon_2$ holds at least with probability $1 - \delta_2$.

$$\left|\ln\frac{P\{C = H \mid X\}}{P\{\widehat{C} = H \mid \widehat{X}\}}\right| = \left|\ln\frac{P\{H - B|X|\}}{P\{H - \widehat{B} \mid \widehat{X}\}}\right|$$
$$= \frac{1}{2\sigma_2^2}\left|\sum_{1\leq i\leq j\leq p}\left[\left(H_{ij} - \widehat{B}_{ij}\right)^2 - \left(H_{ij} - B_{ij}\right)^2\right]\right|$$
$$= \frac{1}{2\sigma_2^2}\left|\sum_{1\leq i\leq j\leq p}\left[2\left(H_{ij} - B_{ij}\right)\left(B_{ij} - \widehat{B}_{ij}\right) + \left(B_{ij} - \widehat{B}_{ij}\right)^2\right]\right|$$

$$\begin{aligned}&\leq \frac{1}{2\sigma_2^2}\sum_{1\leq i\leq j\leq p} 2\left\|\left(H_{ij}-B_{ij}\right)\left(x_{N^{(1)}i}x_{N^{(1)}j}-\hat{x}_{N^{(1)}i}\hat{x}_{N^{(1)}j}\right)\right\| \\ &\quad + \frac{1}{2\sigma_2^2}\sum_{1\leq i\leq j\leq p}\left(x_{N^{(1)}i}x_{N^{(1)}j}-\hat{x}_{N^{(1)}i}\hat{x}_{N^{(1)}j}\right)^2. \end{aligned} \tag{17}$$

By using the Lagrange multiplier method and the inequality in [18], the following inequalities hold:

$$\begin{aligned}&\sum_{1\leq i\leq j\leq p}\left(x_{N^{(1)}i}x_{N^{(1)}j}-\hat{x}_{N^{(1)}i}\hat{x}_{N^{(1)}j}\right)^2 \leq 2, \\ &\sum_{1\leq i\leq j\leq p}\left|\left(x_{N^{(1)}i}x_{N^{(1)}j}-\hat{x}_{N^{(1)}i}\hat{x}_{N^{(1)}j}\right)\right| \leq p+1. \end{aligned} \tag{18}$$

Then, $|\ln ((P\{H-B|X|\})/(P\{H-\hat{B}|\hat{X}\}))|| \leq r(p+1) +1/\sigma_2^2$, where $|G_{ij}| = |H_{ij}-B_{ij}| \leq r$ for all i, j.

The rest of the proof process is similar to Theorem 5, then we can obtain the following:

$$\begin{aligned}&\sigma_2 \geq \frac{(p+1)\sqrt{\ln\left(2/\pi\delta_2^2\right)}+\sqrt{(p+1)^2\ln\left(2/\pi\delta_2^2\right)+4\varepsilon_2}}{2\varepsilon_2}, \\ &0<\delta_2<\sqrt{\frac{2}{\pi}}. \end{aligned} \tag{19}$$

We have proven that the within-class mean vector $\mu^{(k)}$ $(k = 1, 2)$ satisfies $(\varepsilon_1, \delta_1)$ differential privacy, the pooled within-class scatter matrix S_w satisfies $(\varepsilon_2, \delta_2)$ differential privacy, by the property of differential privacy sequential composition, the projection direction vector in the Algorithm 1 satisfies (ε, δ) differential privacy, where $\varepsilon = \varepsilon_1 + \varepsilon_2, \delta = \delta_1 + \delta_2$. For the published projection data $\tilde{X} = X\mathbf{w}$, $X \in R^{N\times p}$, $\mathbf{w} \in R^{p\times 1}$, $p < N$, we can regard $\tilde{X} = X\mathbf{w}$ as a set of undetermined system of equation, the number of variables are more than equations, so the equation has infinitely many sets of solutions, that is, it is impossible to infer the information of the original data X from the published projection data $\tilde{X}$.

4.2. Mul-LDA-DP Algorithm. In this section, we propose the Mul-LDA-DP algorithm for distributed data publishing. The mathematical notations used in this section are summarized in Table 2.

4.2.1. Problem Statement and Algorithm Proposed. In the distributed scenario, data are stored by multiple data owners rather than a single owner, and the data owners do not trust each other. Data at a single site may not be sufficient for statistical learning. One solution is that each data owner uses the LDA-DP algorithm in Section 4.1 to publish the projection data independently. Another solution is the data owners cooperate with each other to publish the projection data of the integrated data. Comparing the two solutions, it is obvious that the latter solution can improve the utility of publishing data. Based on the idea of the second solution and [32], we propose the Mul-LDA-DP algorithm for distributed data publishing. The entity description of the model is as follows.

(1) Data owner. The data owner P_m $(m = 1, 2, \ldots, M)$ has a data set X_m. Each data owner can generate random vectors and matrices to perturb the within-class mean vectors and within-class scatter matrices locally.

(2) Data publisher. The data publisher is a data publishing platform based on blockchain. The data publisher aggregates the local within-class mean vectors and within-class scatter matrices with noise. The data publisher can obtain the projection vector that satisfies differential privacy and publishes the projection data of the pooled data.

(3) A random number generator. It can generate random vectors and random matrices and send them to data owners and data publisher secretly.

Threat Model. In our setting, we assume that the data owners and data publisher are honest-but-curious, that is, they follow the protocol but may try to deduce information of other data owners from the received messages.

Two types of adversaries are considered, which are external attackers and internal attackers. External attackers which can be called an external eavesdropper may gain access to information such as data sent by data owners to the data publisher. Internal adversaries can be the data owners and the data publisher. The goal of each data owner is to extract the information not owned by him, while the goal of the data publisher is to extract the information from each data owner.

Distributed Within-Class Mean Vectors and Pooled Within-Class Scatter Matrix Computation. When the data are owned by M data owners, the within-class mean vectors (1) can be decomposed into the following:

$$\mu^{(k)} = \frac{1}{N^{(k)}}\sum_{m=1}^{M} N_m^{(k)}\mu_m^{(k)}, \quad k = 1, 2, \tag{20}$$

where $\mu_m^{(k)} = 1/N_m^{(k)}\sum_{\mathbf{x}\in X_m^{(k)}}\mathbf{x}$.

The pooled within-class scatter matrix (5) can be decomposed into the following:

$$S_w = \sum_{k=1}^{2} S^{(k)} = \sum_{k=1}^{2}\sum_{m=1}^{M} S_m^{(k)} = \sum_{m=1}^{M}\sum_{k=1}^{2} S_m^{(k)}, \tag{21}$$

where $S_m^{(k)} = \sum_{\mathbf{x}\in X_m^{(k)}}\mathbf{x}\mathbf{x}^{\mathrm{T}} - N_m^{(k)}\mu_m^{(k)}(\mu_m^{(k)})^{\mathrm{T}}$.

The abovementioned result allows each data owner to compute and perturb a partial result simultaneously locally. Therefore, we use the additivity of Gaussian distribution to

Table 2: Summary of notations.

Notation	Explanation
M	The number of data owners
P_m	The m-th data owner
$N_m^{(k)}$	The number of individuals in the k-th class owned by P_m
$N^{(k)}$	The total number of individuals in the k -th class, $N^{(k)} = \sum_{m=1}^{M} N_m^{(k)}$
$X_m^{(k)}$	The set of the k -th class data owned by P_m
$X^{(k)}$	The set of the k -th class data. $X^{(k)} = \cup_{m=1}^{M} X_m^{(k)}$
X_m	The data set owned by P_m. $X_m = \cup_{k=1}^{2} X_m^{(k)}$
$\mu_m^{(k)}$	The within-class mean vector of the k -th class data owned by P_m
$\mu^{(k)}$	The within-class mean vector of the k -th class data
$S_m^{(k)}$	The within-class scatter matrix of the k -th class data owned by P_m
$S^{(k)}$	The within-class scatter matrix of the k-th class data
S_w	The pooled within-class scatter matrix

propose a correlated noise generation scheme. We design the noise generation procedure such that (i) we can ensure that the data output from each data owner satisfy differential privacy and (ii) we can achieve the noise level of the same as the pooled data scenario.

Scheme for Perturbing Shared Data by Correlated Noise. To prevent the data publisher and other data owners learning the privacy of local data, the data owner uses the noise generated by himself and the noise generated by the random number generator to perturb the local within-class mean vectors and within-class scatter matrices. Through our correlated noise design scheme, the data aggregated by the data publisher contain the same level of noise as the centralized scenario. The scheme is described as below:

(1) Initialization stage. The random number generator generates p dimensional random vectors $\overline{\mathbf{g}}_m^{(k)}$, each entry is sampled from $N(0, (M-1)/M\sigma_1^2)$, generates $p \times p$ random matrices $\overline{G}_m$, let $\overline{G}_m$ be the symmetric matrix with the upper triangle (including the diagonal) entries are sampled from $N(0, (M-1)/M\sigma_2^2)$, and makes the symmetrical position entries in the lower triangle matrix equal to the upper triangle, $m = 0, 1, 2, \ldots, M, k = 1, 2$. Make these random vectors and matrices satisfy $\sum_{m=0}^{M} \overline{\mathbf{g}}_m^{(k)} = 0$, $\sum_{m=0}^{M} \overline{G}_m = (0)_{p\times p}$, then $\overline{\mathbf{g}}_m^{(k)}$ $(k = 1, 2)$ and $\overline{G}_m$ are sent to data owner P_m secretly, $\overline{\mathbf{g}}_0^{(k)}$ $(k = 1, 2)$ and $\overline{G}_0$ are sent to the data publisher secretly.

(2) Data owner P_m generates p dimensional random vectors $\mathbf{g}_m^{(k)}$ $(k = 1, 2)$, each entry is sampled from $N(0, 1/M\sigma_1^2)$, computes $\mu_m^{(k)} = 1/N_m^{(k)}(\sum_{\mathbf{x}\in X_m^{(k)}} \mathbf{x} + \mathbf{g}_m^{(k)} + \overline{\mathbf{g}}_m^{(k)})$, $k = 1, 2$, and sends them to the data publisher.

(3) The data publisher computes $\mu^{(k)} = 1/N^{(k)}(\sum_{m=1}^{M} N_m^{(k)} \mu_m^{(k)} + \overline{\mathbf{g}}_0^{(k)})$, $k = 1, 2$ and sends them to each data owner.

(4) The data owner P_m generates $p \times p$ random matrix G_m, let G_m be the symmetric matrix with the upper triangle (including the diagonal) entries are sampled from $N(0, 1/M\sigma_2^2)$, and make the symmetrical position entries in the lower triangle matrix equal to the upper triangle. Data owner P_m computes $S_m = \sum_{k=1}^{2} S_m^{(k)} + G_m + \overline{G}_m$ and sends it to the data publisher.

(5) The data publisher computes $S_w = \sum_{m=1}^{M} S_m + \overline{G}_0$ and calculates the projection vector $\mathbf{w}$ that satisfies differential privacy.

The specific details of Mul-LDA-DP algorithm are in Algorithm 2. The input random vectors $\overline{\mathbf{g}}_m^{(k)}$ and random matrices $\overline{G}_m^{(k)}$ in Algorithm 2 are generated in the initialization stage by the random number generator, $m = 0, 1, 2, \ldots, M, k = 1, 2$.

4.2.2. Privacy Analysis of the Mul-LDA-DP Algorithm

Theorem 7. *The within-class mean vector $\mu^{(k)}$ $(k = 1, 2)$ in Algorithm 2 satisfies $(\varepsilon_1, \delta_1)$ differential privacy.*

Proof. $\mu_m^{(k)} = 1/N_m^{(k)}(\sum_{\mathbf{x}\in X_m^{(k)}} \mathbf{x} + \mathbf{g}_m^{(k)} + \overline{\mathbf{g}}_m^{(k)})$ because each entry of $\mathbf{g}_m^{(k)}$ is sampled from $N(0, 1/M\sigma_1^2)$, and each entry of $\overline{\mathbf{g}}_m^{(k)}$ is sampled from $N(0, M-1/M\sigma_1^2)$, so each entry of $\mathbf{g}_m^{(k)} + \overline{\mathbf{g}}_m^{(k)}$ obeys $N(0, \sigma_1^2)$. By Theorem 5, $\mu_m^{(k)}$ satisfies $(\varepsilon_1, \delta_1)$ differential privacy.

Due to the post-processing property of differential privacy, the within-class mean vector $\mu^{(k)} = 1/N^{(k)}(\sum_{m=1}^{M} N_m^{(k)} \mu_m^{(k)} + \overline{\mathbf{g}}_0^{(k)})$ in Algorithm 2 satisfies $(\varepsilon_1, \delta_1)$ differential privacy.

Theorem 8. *The pooled within-class scatter matrix S_w in Algorithm 2 satisfies $(\varepsilon_2, \delta_2)$ differential privacy.*

Proof. $S_m = \sum_{k=1}^{2} S_m^{(k)} + G_m + \overline{G}_m$, where each entry of symmetric random matrix G_m is sampled from $N(0, 1/M\sigma_2^2)$, and each entry of symmetric random matrix $\overline{G}_m$ is sampled from $N(0, (M-1)/M\sigma_2^2)$, so each entry of $\overline{G}_m + \overline{G}_m$ obeys $N(0, \sigma_2^2)$. By Theorem 6, S_m satisfies $(\varepsilon_2, \delta_2)$ differential privacy. Due to the post-processing property of differential privacy, the pooled within-class scatter matrix $S_w = \sum_{m=1}^{M} S_m + \overline{G}_0$ in Algorithm 2 satisfies $(\varepsilon_2, \delta_2)$ differential privacy.

We have proven both $\mu^{(k)}$ $(k = 1, 2)$ and S_w satisfy differential privacy, we will show that the level of noise is the same as the centralized scenario. In the initialization stage,

```
Input: Data sets $X_m$, $m = 1, 2, \ldots, M, k = 1, 2$, privacy parameters $(\varepsilon_1, \delta_1)$, $(\varepsilon_2, \delta_2)$, random vector $\overline{\mathbf{g}}_m^{(k)}$ and random matrix $\overline{G}_m$ which are generated in initialization stage, $m = 0, 1, 2, \ldots, M; k = 1, 2$.
Output: Projection direction vector $\mathbf{w}$, projection data $\widetilde{X}$
for $m = 1$ to $M$ do
    for $k = 1$ to 2 do
        Set $\sigma_1 = p^{3/2}\sqrt{\ln 2/\pi\delta_1^2} + \sqrt{p^3 \ln 2/\pi\delta_1^2 + 2\varepsilon_1}/\varepsilon_1$, data owner generates $p$ dimensional random vector $\mathbf{g}_m^{(k)}$, each entry is sampled from $N(0, \sigma_1^2/M)$
        Compute $\mu_m^{(k)} = 1/N_m^{(k)}(\sum_{\mathbf{x}\in X_m^{(k)}} \mathbf{x} + \mathbf{g}_m^{(k)} + \overline{\mathbf{g}}_m^{(k)})$
    end for
end for
Compute $\mu^{(k)} = 1/N^{(k)}(\sum_{m=1}^{M} N_m^{(k)}\mu_m^{(k)} + \overline{\mathbf{g}}_0^{(k)})$
for $m = 1$ to $M$ do
    Set $\sigma_2 = (p+1)\sqrt{\ln 2/\pi\delta_2^2} + \sqrt{(p+1)^2 \ln 2/\pi\delta_2^2 + 4\varepsilon_2}/2\varepsilon_2$, data owner generates $p \times p$ symmetric random matrices $G_m$, each entry is sampled from $N(0, \sigma_2^2/M)$
    for $k = 1$ to 2 do
        Compute $S_m^{(k)} = \sum_{\mathbf{x}\in X_m^{(k)}} \mathbf{x}\mathbf{x}^{\mathrm{T}} - N_m^{(k)}\mu_m^{(k)}(\mu_m^{(k)})^{\mathrm{T}}$
    end for
    Compute $S_m = \sum_{k=1}^{2} S_m^{(k)} + G_m + \overline{G}_m$
end for
Compute $S_w = \sum_{m=1}^{M} S_m + \overline{G}_0$
Compute $\mathbf{w} = S_w^{-1}(\mu^{(1)} - \mu^{(2)})$
return $\widetilde{X} = \cup_{m=1}^{M} X_m \mathbf{w}$
```

Algorithm 2: Mul-LDA-DP algorithm.

the noise vectors and matrices generated by the random number generator satisfy $\sum_{m=0}^{M} \overline{\mathbf{g}}_m^{(k)} = 0$ and $\sum_{m=0}^{M} \overline{G}_m = (0)_{p\times p}$.

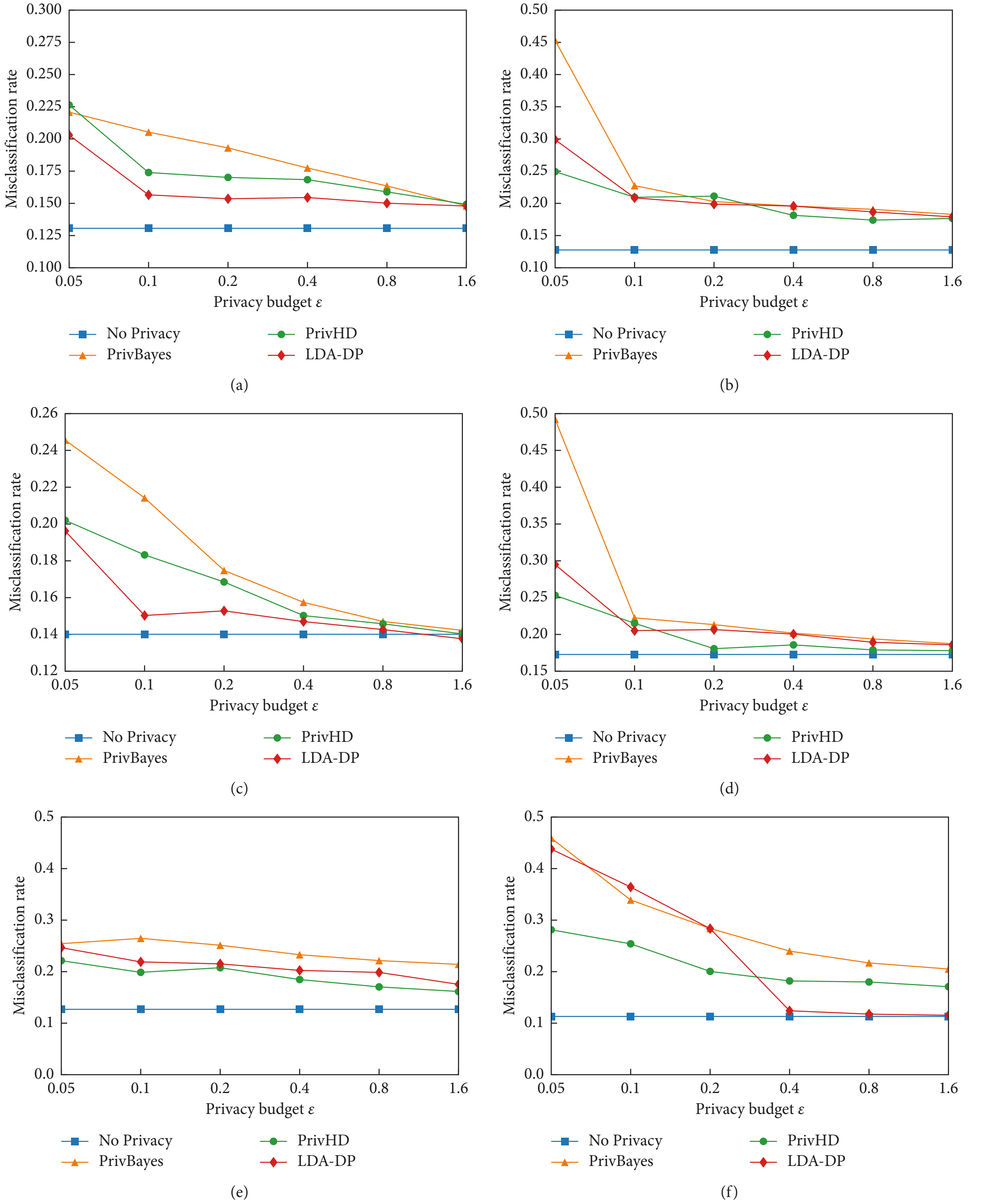
The within-class mean vector $\mu^{(k)}$ $(k = 1, 2)$ is as follows:

$$\begin{aligned}
\mu^{(k)} &= \frac{1}{N^{(k)}}\left(\sum_{m=1}^{M} N_m^{(k)}\mu_m^{(k)} + \overline{\mathbf{g}}_0^{(k)}\right) = \frac{1}{N^{(k)}}\left[\sum_{m=1}^{M}\left(\sum_{\mathbf{x}\in X_m^{(k)}} \mathbf{x} + \mathbf{g}_m^{(k)} + \overline{\mathbf{g}}_m^{(k)}\right) + \overline{\mathbf{g}}_0^{(k)}\right],\\
&= \frac{1}{N^{(k)}}\left[\sum_{m=1}^{M}\sum_{\mathbf{x}\in X_m^{(k)}} \mathbf{x} + \sum_{m=1}^{M} \mathbf{g}_m^{(k)} + \sum_{m=0}^{M} \overline{\mathbf{g}}_m^{(k)}\right] = \frac{1}{N^{(k)}}\left[\sum_{m=1}^{M}\sum_{\mathbf{x}\in X_m^{(k)}} \mathbf{x} + \sum_{m=1}^{M} \mathbf{g}_m^{(k)}\right],\\
&= \frac{1}{N^{(k)}}\sum_{\mathbf{x}\in X^{(k)}} \mathbf{x} + \sum_{m=1}^{M} \mathbf{g}_m^{(k)}.
\end{aligned} \tag{22}$$

Each entry of $\sum_{m=1}^{M} \mathbf{g}_m^{(k)}$ obeys $N(0, \sigma_1^2)$.

The pooled within-class scatter matrix S_w is as follows:

$$\begin{aligned}
S_w &= \sum_{m=1}^{M} S_m + \overline{G}_0 = \sum_{m=1}^{M}\left(\sum_{k=1}^{2} S_m^{(k)} + G_m + \overline{G}_m\right) + \overline{G}_0,\\
&= \sum_{m=1}^{M}\sum_{k=1}^{2} S_m^{(k)} + \sum_{m=1}^{M} G_m + \sum_{m=0}^{M} \overline{G}_m = \sum_{k=1}^{2}\sum_{m=1}^{M} S_m^{(k)} + \sum_{m=1}^{M} G_m,\\
&= \sum_{k=1}^{2} S^{(k)} + \sum_{m=1}^{M} G_m.
\end{aligned} \tag{23}$$



(a)

(b)

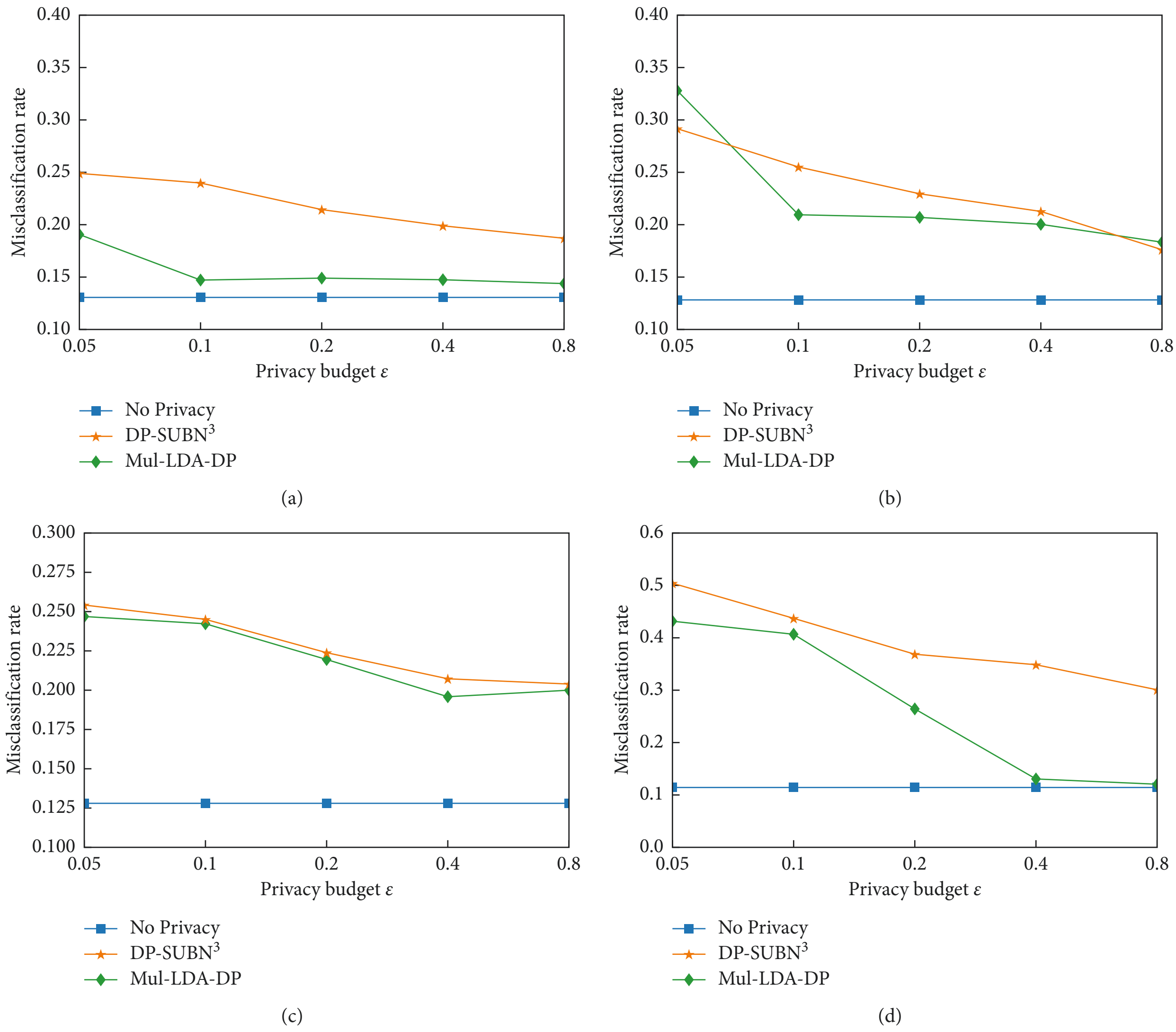
(c)

(d)

(e)

(f)

Figure 1: SVM misclassification rate of LDA-DP, PrivBayes and PRivHD under different privacy budgets. (a) NLTCS, $Y = \text{money}$. (b) NLTCS, $Y = \text{outside}$. (c) NLTCS, $Y = \text{bathing}$. (d) NLTCS, $Y = \text{travelling}$. (e) Adult, $Y = \text{salary}$. (f) Adult, $Y = \text{education}$.



(a)

(b)

(c)

(d)

Figure 2: SVM misclassification rate of Mul-LDA-DA and DP-SUBN^3 under different privacy budgets. (a) NLTCS, Y = money. (b) NLTCS, Y = outside (c) Adult, Y = salary. (d) Adult, Y = education.

Each entry of $\sum_{m=1}^{M} G_m$ obeys $N(0, \sigma_2^2)$.

According to Theorems 5 and 6, the within-class mean vector $\mu^{(k)}$ $(k = 1, 2)$ and pooled within-class scatter matrix S_w contain the same level of noise as the centralized scenario, and we achieve the purpose of improving the utility of publishing data while protecting the data privacy.

There are three opportunities for attackers to steal the data transmitted between the data owner and the data publisher. The first time is that the data owner sends the within-class mean vectors to the data publisher, the second time is that the data owner sends the within-class scatter matrices to data publisher. From Theorems 7 and 8, we know that the within-class mean vectors and the within-class scatter matrices satisfy differential privacy. Therefore, the attacker cannot infer the information of the original data from the eavesdropped data. The third time is that the data owner sends projection data to the data publisher, in Section 4.1.2, we have analyzed that it is impossible to infer the information of the original data from the published projection data.

5. Experiment

In order to measure the usability of the LDA-DP and Mul-LDA-DP algorithms proposed in this paper, we conduct experiments on real data sets which are Adult and NLTCS. Adult data set is extracted from the 1994 US Census, it contains 45222 individuals, each individual has 15 attributes. NLTCS data set is extracted from the National Long Term Care Survey, and recorded the daily activities of 21574 disabled persons at different time periods, each individual has 16 attributes. We use the SVM misclassification rate to measure the availability of the published data. For the Adult data set, it is necessary to predict whether a person (1) holds a post-secondary degree and (2) earns more than 50K. For the NLTCS data set, we need to predict whether a person (1) is unable to get outside, (2) is unable to manage money, (3) is unable to travel, and (4) is unable to bath. In our experiments, we set $\delta = 0.001$ to remain unchanged, and ε to take different values. We uniformly divide the privacy parameters

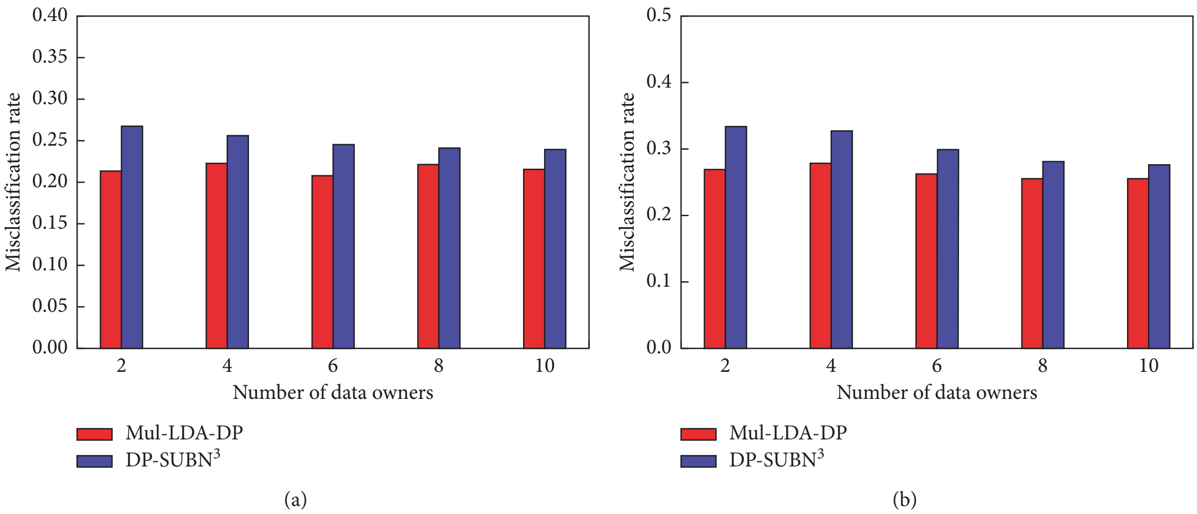


Figure 3: SVM misclassification rate of Mul-LDA-D and DP-SUBN^3 under different number of data owners. (a) Adult, Y = salary. (b) Adult, Y = education.

into 2 portions ($\varepsilon_1 = \varepsilon_2 = \varepsilon/2, \delta_1 = \delta_2 = \delta/2$). Each experiment was repeated 50 times, and the mean value was taken as the experimental result. We use "No Privacy" to represent the SVM misclassification rate on the original data set.

5.1. Comparing the Performance of LDA-DA, PrivBayes, and PRivHD Algorithms under Different Privacy Budgets. The LDA-DA, PrivBayes, and PrivHD algorithms are all suitable for the centralized data publishing scenario, so in this set of experiments, we set the number of data owners to 1, and privacy budget ε takes different values. As can be seen from Figure 1, for both Adult and NLTCS data sets, the SVM classification utility of the data published by the LDA-DP algorithm outperforms the PrivBayes algorithm. The LDA-DP algorithm outperforms the PrivHD algorithm on the NLTCS dataset; however, the LDA-DP algorithm has slightly lower SVM classification utility on the Adult dataset than the PrivHD algorithm. We can also observe a commonality, for LDA-DA, PrivBayes, and PRivHD algorithms, the SVM misclassification rate decreases with the increase of the privacy budget ε. This phenomenon is consistent with the theory that as the privacy budget ε increases, privacy protection will weaken and the availability of data will increase.

5.2. Comparing the Performance of Mul-LDA-DA and DP-SUBN^3 Algorithms under Different Privacy Budgets. The algorithm Mul-LDA-DP proposed in this paper is suitable for the distributed data publishing scenario, so in this set of experiments, we set the number of data owners to 3, and privacy budget ε takes different values. We train classifiers on published data set to compare the efficacy of Mul-LDA-DA and DP-SUBN^3 algorithms. From Figure 2, we can see that the SVM classification utility of the data published by the Mul-LDA-DP algorithm outperforms the DP-SUBN^3 algorithm. Both on money of NLTCS and education of Adult classifiers, the misclassification rate of Mul-LDA-DA algorithm is significantly lower than the DP-SUBN^3 algorithm especially.

5.3. Comparing the Performance of Mul-LDA-DA and DP-SUBN^3 Algorithms under Different Number of Data Owners. In this section, the experiment studied the relationship between SVM misclassification rate and the number of data owners. The number of data owners is set to 2, 4, 6, 8, 10, and the privacy budget ε is set to 0.2, We trained two classifiers, education classifier, and salary classifier on Adult data set. The results in Figure 3 show that the SVM misclassification rate of the Mul-LDA-DP algorithm remains stable with the change of the number of data owners. The reason is that we perturb the local shared data by generating correlated noise based on the additivity of the Gaussian distribution. This scheme ensures that the level of Gaussian noise added to the data in the distributed scenario is similar to the noise level in the centralized scenario. Therefore, as the number of data owners increases, the misclassification rate remains stable. The SVM misclassification rate of DP-SUBN^3 algorithm decreases as the number of data owners increases. This is because as the number of data owners increases, the number of update iterations increases when constructing the Bayesian network, and the Bayesian network constructed is closer to the distribution of the original data. However, from Figure 3, we can see that the performance of Mul-LDA-DA algorithm is still better than DP-SUBN^3 algorithm when the number of data owners is no more than 10.

6. Conclusion

In this paper, we propose two algorithms for privacy preserving data publishing, the LDA-DP algorithm for data publishing in the scenario, and the Mul-LDA-DP algorithm for multiparty horizontally split data publishing. We use the additivity of Gaussian distribution to alleviate the effects of noise and can achieve the same noise level as the centralized scenario. The experimental results show that the projection data released by the two algorithms can maintain high utility in SVM classification. However, the research in this paper also has limitations. 1)We only research the privacy protection problem when the data are a binary classification, but they are often multiclassification data. 2)The data released by the two algorithms in this paper are low-dimensional projection data of the original data, which limit the analysis and mining of the released data in many aspects. In the future, we will continue to conduct research on the abovementioned issues.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] Y. Chen, J. Sun, Y. Yang, T. Li, X. Niu, and H. Zhou, “Psspr: a source location privacy protection scheme based on sector phantom routing in wsns,” *International Journal of Intelligent Systems*, vol. 37, no. 2, pp. 1204–1221, 2021.

[2] Q. Liu, J. Yu, J. Han, and X. Yao, “Differentially private and utility-aware publication of trajectory data,” *Expert Systems with Applications*, vol. 180, no. 7, Article ID 115120, 2021.

[3] L. Sweeney and L. K. Anonymity, “A model for protecting privacy,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, 2002.

[4] R. Rongxing Lu, X. Xiaohui Liang, L. Xu Li, X. Xiaodong Lin, and X. Xuemin Shen, “Eppa: an efficient and privacy-preserving aggregation scheme for secure smart grid communications,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pp. 1621–1631, 2012.

[5] C. Wang, D. Wang, G. Xu, and D. He, “Efficient Privacy-Preserving User Authentication Scheme with Forward Secrecy for Industry 4.0,” *SCIENCE CHINA: Information Sciences*, vol. 65, Article ID 112301, 2020.

[6] C.-T. Li, D.-H. Shih, C.-C. Wang, C.-L. Chen, and C.-C. Lee, “A blockchain based data aggregation and group authentication scheme for electronic medical system,” *IEEE Access*, vol. 8, Article ID 173904, 2020.

[7] T. Li, Z. Wang, Y. Chen, C. Li, Y. Jia, and Y. Yang, “Is semi-selfish mining available without being detected?” *International Journal of Intelligent Systems*, vol. 36, 2021.

[8] T. Li, Z. Wang, G. Yang, Y. Cui, Y. Chen, and X. Yu, “Semi-selfish mining based on hidden Markov decision process,” *International Journal of Intelligent Systems*, vol. 36, no. 7, pp. 3596–3612, 2021.

[9] Y.-T. Tsou and B.-C. Lin, “PPDCA: privacy-preserving crowdsourcing data collection and analysis with randomized response,” *IEEE Access*, vol. 6, Article ID 76970, 2018.

[10] X. Ren, C.-M. Yu, W. Yu et al., “High-dimensional Crowdsourced Data Publication with Local Differential Privacy,” *IEEE Transactions on Information Forensics & Security*, vol. 13, no. 9, pp. 2151–2166, 2018.

[11] Y. Chen, D. Sen, T. Li, Y. Wang, and H. Zhou, “Dynamic multi-key fhe in asymmetric key setting from lwe,” *IEEE Transactions on Information Forensics and Security*, vol. 16, no. 1–1, 2021.

[12] Q. Wang, Y. Zhang, L. Xiao, Z. Wang, and K. Ren, “Rescuedp: real-time spatio-temporal crowd-sourced data publishing with differential privacy,” in *Proceedings of the IEEE Infocom -the IEEE International Conference on Computer Communications*, San Francisco, CA, USA, April, 2016.

[13] W. Hao and Z. Xu, “Cts-dp: publishing correlated time-series data via differential privacy,” *Knowledge-Based Systems*, vol. 122, pp. 167–179, 2017.

[14] H. Wang and H. Wang, “Correlated tuple data release via differential privacy,” *Information Sciences*, vol. 560, no. 347–369, 2021.

[15] S. Chen, A. Fu, S. Yu, H. Ke, and M. S. Dp-qic, “A differential privacy scheme based on quasi-identifier classification for big data publication,” *Soft Computing*, vol. 25, no. 3, 2021.

[16] Z. Gu, G. Zhang, and C. Yang, “Multi-party high-dimensional related data publishing via probabilistic principal component analysis and differential privacy,” in *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, W. Shi, X. Chen, and KK. R. Choo, Eds., Springer International Publishing, Cham, Switcher land, pp. 117–131, 2022.

[17] K. Nissim, F. D. Mcsherry, C. Dwork, and A. L. Blum, “Practical privacy: the sulq framework,” in *Proceedings of the Twenty-Fourth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, Baltimore, Maryland, USA, June, 2005.

[18] K. Chaudhuri, A. D. Sarwate, and K. Sinha, “A near-optimal algorithm for differentially-private principal components,” *Journal of Machine Learning Research*, vol. 14, pp. 2905–2943, 2013.

[19] J. Zhang, G. Cormode, C. M. Procopiuc, D. Srivastava, and X. Xiao, “Privbayes: private data release via bayesian networks,” *ACM Transactions on Database Systems*, vol. 42, no. 4, pp. 1–41, 2014.

[20] C. Rui, X. Qian, Z. Yu, and J. Xu, “Differentially private high-dimensional data publication via sampling-based inference,” in *Proceedings of the 21th ACM SIGKDD International Conference*, Sydney, Australia, August, 2015.

[21] X. Zhang, L. Chen, K. Jin, and X. Meng, “Private High-Dimensional Data Publication with junction Tree,” *Journal of Computer Research and Development*, vol. 55, no. 12, 2018.

[22] C. Xu, J. Ren, Y. Zhang, Z. Qin, and K. Ren, “Dppro: Differentially Private High-Dimensional Data Release via Random Projection,” *IEEE Transactions on Information Forensics and Security*, vol. 1299 pages, 2017.

[23] W. Zhang, J. Zhao, F. Wei, and Y. Chen, “Differentially private high-dimensional data publication via Markov network,” *ICST Transactions on Security and Safety*, vol. 6, no. 19, Article ID 159626, 2019.

[24] J. Ge, Z. Wang, M. Wang, and L. Han, *Minimax-optimal Privacy-Preserving Sparse Pca in Distributed Systems*, in *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistic*, Lanzarote, Canary Islands, April, 2018.

[25] S. Wang and J. M. Chang, “Differentially private principal component analysis over horizontally partitioned data,” in *In Proceedings of the 2018 IEEE Conference on Dependable and Secure Computing (DSC)*, Kaohsiung, Taiwan, December, 2018.

[26] H. Imtiaz and A. D. Sarwate, “Differentially private distributed principal component analysis,” in *Proceedings of the ICASSP 2018-2018 IEEE International Conference on*

Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, April, 2018.

[27] D. Alhadidi, N. Mohammed, B. Fung, and M. Debbabi, *Secure Distributed Framework for Achieving -differential Privacy*. Springer, Berlin, Heidelberg, 2012.

[28] X. Cheng, P. Tang, S. Su, R. Chen, Z. Wu, and B. Zhu, "Multiparty high-dimensional data publishing under differential privacy," *IEEE Transactions on Knowledge and Data Engineering*, vol. 1–1, 2019.

[29] R. Wang, B. Fung, Y. Zhu, and Q. Peng, "Differentially private data publishing for arbitrarily partitioned data," *Information Sciences*, vol. 553, no. 10, 2020.

[30] C. Dwork, K. Kenthapadi, M. Frank, I. Mironov, and M. Naor, *Our Data, Ourselves: Privacy via Distributed Noise Generation*, DBLP, Trier, Germany, 2006.

[31] Cynthia, A. Dwork, and Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, 2013.

[32] H. Imtiaz and A. D. Sarwate, "Distributed differentially-private algorithms for matrix and tensor factorization," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, pp. 1449–1464, 2018.