Computers & Security

UMAP_PM: Local Differential Private High-Dimensional Data Release via UMAP

--Manuscript Draft--

Manuscript Number:	COSE-D-23-01223
Article Type:	Full Length Article
Keywords:	Privacy protection High-dimensional data Local differential privacy Dimensionality reduction machine learning
Corresponding Author:	Xuebin Ma Inner Mongolia University CHINA
First Author:	Aixin Lin
Order of Authors:	Aixin Lin
	Xuebin Ma
Abstract:	<div>Protecting the privacy of high-dimensional datasets while releasing representative data has been a</div><div>focus of much attention in recent years. Local differential privacy (LDP) has emerged as a promising</div><div>privacy standard for mining and releasing data. However, most existing research focuses on applying</div><div>LDP to complex data and/or analysis tasks, leaving the fundamental problem of releasing high</div><div>dimensional data under LDP insufficiently addressed. Applying LDP to high-dimensional data poses</div><div>significant challenges due to the increased risk of perturbation error and computational complexity.</div><div>Motivated by this, we propose a novel LDP mechanism named UMAP_PM, which leverages manifold</div><div>and projection techniques to refine complex relationships between attributes and provide appropriate</div><div>LDP for each result vector. By balancing data utility and privacy, and reducing computational</div><div>time, UMAP_PM maximizes the utility of data while ensuring privacy. We experimentally evaluate</div><div>UMAP_PM on real data and demonstrate its superiority over existing solutions in terms of accuracy</div><div>and time complexity.&nbsp;</div>

June 5th, 2023

Dear Editors:

We are pleased to submit China Communications for publication. Our paper is entitled “UMAP_PM: Local Differential Private High-Dimensional Data Release via UMAP” by Aixin Lin and Xuebin Ma. An electronic version of the manuscript has been sent to you via the website of the journal. I would like to declare on behalf of my co-authors that the work described is original research that has not been published previously, and not under consideration for publication elsewhere.

The purpose of this paper is to protect the privacy of published high-dimensional data. Protecting the privacy of high-dimensional datasets while releasing representative data has been a focus of much attention in recent years. Local differential privacy (LDP) has emerged as a promising privacy standard for mining and releasing data. However, most existing research focuses on applying LDP to complex data and/or analysis tasks, leaving the fundamental problem of releasing high-dimensional data under LDP insufficiently addressed. Applying LDP to high-dimensional data poses significant challenges due to the increased risk of perturbation error and computational complexity. Motivated by this, we propose a novel LDP mechanism named UMAP_PM, which leverages manifold and projection techniques to refine complex relationships between attributes and provide appropriate LDP for each result vector. By balancing data utility and privacy, and reducing computational time, UMAP_PM maximizes the utility of data while ensuring privacy. We experimentally evaluate UMAP_PM on real data and demonstrate its superiority over existing solutions in terms of accuracy and time complexity.

I provide my address, phone number, and e-mail address to facilitate your communication:
Aixin Lin
Inner Mongolia Key Laboratory of Wireless Networking and Mobile Computing
Inner Mongolia University
Hohhot 010021
P. R. of China
Tel.: +86-176-4738-2437 (M)
E-mail: linaixin210@163.com

Thank you very much for your attention. Should you have any questions, please feel free to contact me.
Sincerely,
Aixin Lin
Inner Mongolia Key Laboratory of Wireless Networking and Mobile Computing
Inner Mongolia University

Hohhot 010021
P. R. of China

Biographical Sketch

Aixin Lin: is a master’s student at the School of Computer Science, Inner Mongolia University. Her main research interests are privacy preservation and high-dimensional data, focusing on privacy preservation in local differential privacy, high-dimensional data and machine learning.

Xuebin Ma: is an associate professor in the School of Computer Science at Inner Mongolia University. He mainly researches on big data analytics techniques, focusing on federated learning, privacy protection, and secure data sharing and utilization. He has also conducted research in areas such as wireless networks and delay-tolerant networks. In addition, he also conducts research on collaborative topics in the field of information security.

UMAP_PM: Local Differential Private High-Dimensional Data Release via UMAP

Aixin Lin, Xuebin Ma*

Inner Mongolia Key Laboratory of Wireless Networking and Mobile Computing, Inner Mongolia University, Hohhot, China

ARTICLE INFO

Keywords:
Privacy protection
High-dimensional data
Local differential privacy
Dimensionality reduction

ABSTRACT

Protecting the privacy of high-dimensional datasets while releasing representative data has been a focus of much attention in recent years. Local differential privacy (LDP) has emerged as a promising privacy standard for mining and releasing data. However, most existing research focuses on applying LDP to complex data and/or analysis tasks, leaving the fundamental problem of releasing high-dimensional data under LDP insufficiently addressed. Applying LDP to high-dimensional data poses significant challenges due to the increased risk of perturbation error and computational complexity. Motivated by this, we propose a novel LDP mechanism named UMAP_PM, which leverages manifold and projection techniques to refine complex relationships between attributes and provide appropriate LDP for each result vector. By balancing data utility and privacy, and reducing computational time, UMAP_PM maximizes the utility of data while ensuring privacy. We experimentally evaluate UMAP_PM on real data and demonstrate its superiority over existing solutions in terms of accuracy and time complexity.

1. Introduction

Nowadays, with the increasing prevalence of high dimensional data in various fields (Zhong et al., 2023; Zhao et al., 2023; Kabir et al., 2023), such as genomics, neuroimaging, and financial markets, the information from all aspects can be collected and analyzed among various data attributes to better produce rich knowledge, thus benefiting everyone in the high-dimensional data. However, the privacy of participants can be easily inferred or identified due to the release of high-dimensional data, despite the use of privacy-preserving schemes. In recent years, incidents such as the ERFS security flaw in Japan [1] and the Ant Yeah Hey app in China [2] have highlighted the importance of protecting high-dimensional personal data to ensure privacy and security.

Protecting privacy in high-dimensional personal data requires special attention due to high-dimensional data (data with multiple attributes), a lot of potential information and rules behind the data can be mined or extracted to provide accurate dynamics and reliable prediction for both groups and individuals. One potential solution is the use of privacy-preserving data analysis techniques, such as secure multiparty computation and homomorphic encryption. These techniques allow multiple parties to perform data analysis on encrypted data without revealing the data's contents to any individual party. However, when data is limited to a few individuals, traditional methods of releasing high-dimensional data may be computationally expensive due to the increased dimensionality of the data.

Differential privacy (DP) is another approach to protecting privacy in high-dimensional personal data that provides strong guarantees for protecting individual privacy in the release and analysis of such data. This method involves adding random noise to the data to prevent individual identification while maintaining the overall utility of the data.

Centralized Differential Privacy (CDP) (Xiong et al., 2020) is the most widely utilized approach in privacy-preserving algorithms for high-dimensional data. CDP has been mathematically demonstrated to effectively thwart attacks based on background knowledge (Chang et al., 2023; Dwork et al., 2006; Wang et al., 2023; Redberg et al., 2023; Zhang et al., 2023) by adding noise through Gaussian, Laplace, and exponential mechanisms. However, CDP focuses on the assumption of a trustworthy server to synthesize and release high-dimensional data, which may not be feasible in reality due to an insufficient number of reliable entities. And that despite the privacy protection against difference and inference attacks from aggregate queries, high-dimensional data may still suffer from privacy leakage before aggregation because of no guaranteed local privacy on the user side.

Furthermore, the curse of dimensionality, where computational complexity increases exponentially with dimensionality growth, contributes to a substantial computational expense in releasing high-dimensional data. With increasing data dimension, some existing privacy techniques such as differential privacy (a de-facto standard privacy paradigm) become vulnerable when applied directly to multiple highly correlated attributes, increasing the success rate of many reference attacks such as cross-checking, making the released data less useful. Consequently, addressing the curse of dimensionality and non-local privacy are two critical concerns that require immediate attention in privacy-preserving algorithms for high-dimensional data dissemination.

Targeting the dimensional curse problem, various existing schemes demonstrated their effectiveness in terms of different perspectives, which mainly include the following

*Corresponding author
linaixin210@163.com (A. Lin); csmaxuebin@imu.edu.cn (X. Ma)
ORCID(s):

[1] https://www.chinanews.com.cn/gj/2022/03-03/9690843.shtml

[2] https://new.qq.com/omn/20210416/20210416A0E4G600.html

aspects: (1) One approach to dealing with the curse of dimensionality is to break down high-dimensional data into a set of lower-dimensional marginal tables, primarily using Bayesian networks, branch trees, and Markov networks, and to deduce the mechanism of the joint data distribution from the marginal tables (Zhang et al., 2019; Wu et al., 2023; Lu et al., 2022; Zhao et al., 2023; Zhang et al., 2017; Wang et al., 2018; Zhang et al., 2018; Yang et al., 2017; Wang et al., 2022; Xue et al., 2021; Wei et al., 2018). This is a method to integrate the mechanism of the joint distribution of the data into the mechanism of the CDP. However, this method will lead to useless released data in real life, as the probability distribution of high-dimensional data is often unknown. (2) High-dimensional data can be reduced to a lower dimension before privacy protection is applied by using machine learning techniques such as Principal Component Analysis (PCA) (Wang and Xu, 2020; Ge et al., 2018) and Random Projection (Xu et al., 2017). However, the dependencies between high-dimensional data are not taken into account by these methods. As the number of attributes increases, there will be some errors in the learning of the correlations between all the attributes, which will significantly reduce the utility of the data. In addition, these algorithms use CDP, which can lead to the leakage of sensitive data if the central server is untrustworthy.

LDP has been introduced to tackle non-local privacy concerns. The primary distinction between LDP and CDP lies in the absence of a need for a trusted third-party server. With LDP, users can safeguard their privacy at a local level, preventing the central server from accessing the original data, thus enhancing privacy. LDP adheres to the stringent privacy guarantees of DP, ensuring that an adversary (including the aggregator in LDP) cannot infer personal sensitive information with high certainty, regardless of their background knowledge. Nonetheless, high-dimensional data exhibit intricate relationships between attributes. If these dependency relationships are not managed appropriately, the utility of LDP-protected high-dimensional data will be significantly compromised, and run time will increase substantially.

This section discusses the challenge of improving the utility of the data while reducing the running time required for the release of high-dimensional data under data protection constraints. To address the deficiency of the existing methods, we present UMAP_PM, a local differentially private method for releasing high-dimensional data, which makes the high-dimensional under privacy protection have original dependencies. UMAP_PM aims to achieve a balance between privacy and utility while minimizing computational time. Our major contributions are summarized as follows.

- In this paper, we propose a novel mechanism named UMAP_PM which leverages LDP for high-dimensional data release. UMAP_PM employs manifold learning and projection techniques to achieve dimensionality reduction while maintaining the original complex dependencies between data. This mechanism significantly reduces the time required for releasing high-dimensional data while ensuring privacy and data utility.

- In this paper, we propose a strategy for applying LDP to each resulting vector in an optimal way, in order to achieve a balance between data utility and privacy. By doing so, we can ensure that the privacy of the participants is protected while also allowing for the maximum use of the released data.

- In this paper, we implement and evaluate our schemes on different real-world datasets, experimental results show that UMAP_PM achieves significantly better accuracy and lower running time than several state-of-the-art solutions for generating synthetic data.

The remaining sections of this article are organized as follows. Section 2 reviews related work on DP for high-dimensional data release. Section 3 introduces some relevant background knowledge and the definition of LDP. Section 4 provides a detailed description of the UMAP_PM algorithm process. Section 5 analyzes and summarizes the experimental results. Finally, Section 6 concludes the article.

2. Related work

In recent years, differential privacy is a strong privacy standard that provides semantic, information-theoretic guarantees on individuals' privacy, which has attracted much attention from various fields, including high-dimensional data. To balance the demands of privacy and data utility, the development of privacy-preserving techniques for releasing high-dimensional data has become increasingly important. DP for statistical information gathering and estimation has been widely studied, but currently, centralized differential privacy (CDP) is still the most widely used approach in privacy-preserving algorithms for high-dimensional data. There are still relatively few algorithms that use LDP for high-dimensional data publishing. Therefore, we will provide an analysis and summary of the current research status of CDP and LDP in high-dimensional dataset release. Specifically, the release of high-dimensional data based on CDP and LDP can be mainly divided into two aspects: low-dimensional marginal tables and machine learning methods (Wang and Xu, 2020; Ge et al., 2018; Gu et al., 2021; Xu et al., 2017). A full survey of methods to realize differential privacy is beyond the scope of this work. Here, we identify the most related efforts and discuss why they cannot fully solve the problems above.

2.1. Differential Privacy Protection Algorithm Based on Low Dimensional Marginal Table

The most widely used method for inferring the joint data distribution of high-dimensional data via marginal tables is to decompose high-dimensional data into low-dimensional marginal tables. At present, the most commonly used methods are Bayesian networks, decision trees, and Markov networks. Zhang et al. (Zhang et al., 2017) proposed to calculate the mutual information of attribute-parent pairs and then model the correlations via a Bayesian network, it applies the Laplace mechanism to add noise to the constructed Bayesian network to secure CDP. However, in the construction of the Bayesian network, the random selection of initial nodes can lead to uncertainty in the construction of the Bayesian network. Furthermore, this method uses the exponential mechanism for the selection of attribute pairs, and as the number of attribute pairs gradually increases, the selection accuracy will decrease significantly. Zhang et al. (Zhang et al., 2018) proposed a CDP algorithm based on a joint tree for releasing high-dimensional data (PrivHD). This algorithm uses a Markov network to reduce the dimensionality of high-dimensional data and combines it with high-pass filtering to filter the generated attribute pairs, and finally, a joint tree generates a complete clique. This method has obvious utility in querying and classifying while maintaining the privacy provided by CDP algorithms. However, this approach depends on a trusted third-party server to perform CDP, introducing the risk of privacy breaches. Ren et al. in Wang et al. (2022) proposed LoCop and DR_LoCop, which use joint trees for data correlation and trees for data correlation and LDP for the privacy of high-dimensional data. However, the run time of the above methods increases significantly with the data dimension due to the estimation of multi-dimensional joint distributions. The joint distribution of the data is often not known, which makes these methods very limited in the real world. Many dimensionality reduction methods based on machine learning have been used to overcome this limitation.

2.2. Differential Privacy Protection Algorithm Based on Machine Learning

PCA is a representative dimensionality reduction method in machine learning. Wang et al. (Wang and Xu, 2020) proposed a non-interactive LDP-PCA model that uses LDP to address the privacy leakage problem in the high-dimensional data release. However, this algorithm is not accurate in classifying the following. Gu et al. (Gu et al., 2021) proposed a data release method with probabilistic principal component analysis (PPCA), which generates high-dimensional data that satisfies the CDP by using the generative model of PPCA, the accuracy of SVM classification is significantly improved by this algorithm. The goal of PCA is to maximize the projected variance, but when the data dimension is very high, the time complexity of the algorithm increases significantly. Each principal component after PCA is a linear combination of the original variables and is independent of each other, so it fails to reflect the complex dependencies between original high-dimensional data, which greatly reduces the utility of the data.

Random projection (RP) achieves dimensionality reduction by randomly selecting a projection matrix, which has computational advantages over PCA. Xu et al. (Xu et al., 2017) proposed DPPro algorithm satisfying CDP. This method uses RP for dimensionality reduction, preserves L2 distance between users, and adds Gaussian noise to low-dimensional data to guarantee privacy and utility. However, since this method relies on a trusted third-party server, it still risks compromising privacy. Sun et al. (Sun et al., 2020) proposed a high-dimensional numerical data collection algorithm that satisfies the LDP, which is called Multi-RPHM. In this method, the user uses a random projection to reduce the dimensionality of the original high-dimensional data and sends it to the data collector, which then performs the dimensionality reduction and generates a high-dimensional perturbed dataset for the data release. The above algorithms all use RP to reduce the dimensionality of high-dimensional datasets. The principle of random projection is that a random matrix is multiplied by the high-dimensional dataset to reduce its dimensionality. So, when there are too many dimensions in the data, it is more computationally efficient than PCA. However, the data are not meaningful in certain scenarios, as the projection matrix is chosen randomly.

The fuzzy rough set model is a mathematical tool that removes redundant attributes and reduces the dimensionality of data by assuming that there is some indistinguishable relationship between data, which effectively reflects the dependencies between high-dimensional data. Li et al. (Li et al., 2019) proposed a method called DPRers, which combines CDP and rough set for medical dataset privacy. This algorithm uses the Laplace mechanism to add noise to the data mining process, making data very useful. However, if the central server is untrustworthy, the algorithm can lead to serious privacy violations because it uses CDP for data protection.

3. Background

3.1. Local Differential Privacy

In general, the data aggregation server is assumed to be trustworthy in DP. However, it is possible that the server is not trustworthy and is vulnerable to some insider attacks. For this reason, LDP has been proposed, LDP guarantees the privacy of each individual's data on the user side. LDP is formally defined below.

Definition 1 (($\varepsilon -$ LDP)). *(Qin et al., 2016; Fanti et al., 2016) Given N users, each user has a record and is given a privacy algorithm M, if algorithm M obtains the same output result s^* on any two records s and $s^{'}$, the following relationship exists (1), algorithm M is said to meet $\varepsilon -$*LDP*:*

$$\Pr\left(M(s) = s^*\right) \leq e^{\varepsilon} \times \Pr\left(M\left(s'\right) = s^*\right) \tag{1}$$

LDP has two fundamental composition theorems, namely sequential composition and parallel composition (Xu et al., 2020; Wang et al., 2019; Erlingsson et al., 2014; Wang et al., 2018; Zhang et al., 2020; Kim et al., 2020; Wang et al., 2017; Li et al., 2020).

Theorem 1. *Sequence Combination Property: If a user dataset D and l privacy-preserving algorithms $M_1, M_2, \ldots, M_l$ are given, where any one of the algorithms M_i satisfies $\varepsilon_i - LDP$, then the sequence combination of the algorithms $M_1, M_2, \ldots, M_l$ also satisfies $\varepsilon-$*LDP *with a privacy budget $\varepsilon = \sum_{i=1}^{l} \varepsilon_i$.*

Theorem 2. *Parallel Composition Property: If a user dataset D is partitioned into l disjoint subsets, $D = \{D_1, D_2, \ldots, D_l\}$, and any algorithm M satisfies $\varepsilon -$* LDP*, we say that algorithm M satisfies $\varepsilon -$* LDP *on the user dataset $D = \{D_1, D_2, \ldots, D_l\}$, where the privacy budget is ε.*

These theorems provide powerful tools for LDP, enabling developers to choose the most appropriate privacy mechanism based on their specific needs and scenarios. Specifically, the server can apply a series of LDP and allocate a portion of the privacy budget to each mechanism. This series of mechanisms satisfy $\varepsilon -$ LDP.

4. Method

4.1. The Overview of UMAP_PM

For better handling of complex dependencies between high-dimensional data, this section proposes a UMAP_PM algorithm based on the LDP. Fig.1 shows the framework of UMAP_PM. In order to better handle the complex dependencies between high-dimensional data and to make the synthesized data set more effective, the basic idea of the algorithm is as follows: First, the high-dimensional data is pre-processed and transformed into $n \times d$ dimensional data; Second, based on Local Manifold Approximations (LMA) and Local Fuzzy Simplex Set Representation (LFSSR) in topology theory, the local weighted graph is computed and the initial high-dimensional graph is constructed; Third, the spectral embedding dimension reduction is performed. To make the reduced low-dimensional graph as structurally similar as possible to the original high-dimensional graph, cross-entropy and random gradient descent are used to update the low-dimensional graph; Fourth, LDP is used to protect the privacy of the low-dimensional data and prevent sensitive information from being leaked. The detailed steps for UMAP_PM are described in the following:

- Input high-dimensional dataset $X(n \times d)$, reduce to dimension k, control loop optimisation n_epochs, control layout parameters in the algorithm min_dist and privacy budget ε.

- Compute the local fuzzy simple set for each sample in the dataset, and link all local fuzzy simple sets to obtain a high-dimensional topological representation weighted matrix top-rep.

- Optimize the graph layout, embed the top-rep spectrum in high-dimensional space to obtain P', and optimize P' to obtain a dimensionally reduced P.

- Local differential privacy protection for data P, obtain high-dimensional data P^*, with local differential privacy and release it.

Algorithm 1 Local Fuzzy Simplical Set

```
Input: Dataset $X(n \times d)$, $\rho$, $\vartheta$, nearest_numb
Output: Fuzzy Simplical Set: $fs\_set$
Initialize: $fs\_set_0 \leftarrow X$
$fs\_set_1 \leftarrow \{(([d_i, d_j], 0) \mid (d_i, d_j \in X))\}$
for all $d_j \in$ nearest_numb do
    Update nearest neighbor distance:
    $\text{dist}_{d_i,d_j} \leftarrow \max\{0, \text{dist}(d_i, d_j) - \rho\} / \vartheta$
end for
$fs\_set \leftarrow fs\_set\, t_1 \cup ([d_i, d_j], \exp(- \text{dist}_{d_i,d_j}))$
return $fs\_set$
```

4.2. Computing Local Fuzzy Simplicial Set

This section mainly introduces local fuzzy simplicial set computation. Learning the manifold structure in high-dimensional space is a precondition for dimensionality reduction of high-dimensional data while computing local fuzzy simplicial sets is a necessary condition for learning the manifold structure. The local client pre-processes the high-dimensional data and converts it into $n \times d$ dimensional data before the computation of the local fuzzy simplicial set. The specific steps are shown in Algorithm 1.

- Initialize to randomly select a center from the dataset and connect one of its neighbor nodes to the center to obtain $fs\_\ \text{set}_1$.

- For all $d_j \in$ nearest_numb update the distance between the node and the neighbor node, which is determined by the selection of a distance threshold and the determination of the connected component. The distance threshold is used to determine which data points and centers are in the same local region and is a super-parameter that can be adjusted based on the characteristics of the dataset. When selecting the connectivity components, a DFS (Depth-First-Search) based algorithm is used to determine local fuzzy simplicial sets between all data points whose distance from the center point is less than the threshold value. Each local fuzzy simplicial set can be considered as a group containing multiple data points.

- A local weight map is constructed for each group, a distance measuring method is used to calculate the distance between all data points, and the distance is converted into similarity. Using these similarities, a local weight graph is constructed, where the weight between K nearest neighbors of each data point is non-zero, and the remaining weights are zero.

- Sparsely processes the local weight map to reduce computational complexity and the effect of noise.

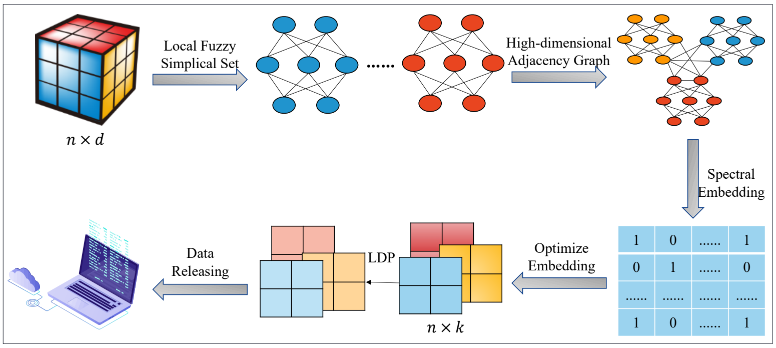


Figure 1: The Algorithm flow chart.

The method used to measure the distance for each local fuzzy simplicial set is not universal in the whole space, but varies between different regions. The distance formula uses ρ and ϑ to make adjustments, ρ is a kernel index that can locally adjust each data point, allowing different distances between data nodes, ensuring local liquidity of manifold structure. Its formula is as follows.

$$\rho = \min \left\{ \operatorname{dist}\left(d_i, d_j\right) \mid 1 \leq j \leq k, \operatorname{dist}\left(d_i, d_j\right) > 0 \right\} \quad (2)$$

After obtaining the local fuzzy simple sets, it is necessary to connect them to obtain the top_rep. The connectivity between the local fuzzy simple sets is determined by the edge weights $\left(\text{edge\_}w_{ij}\right)$, but due to the use of ρ, the different distances that each data point may have will inevitably lead to asymmetric edge weights. For example, the edge weight of point $E \rightarrow F$ is different from the edge weight of point $F \rightarrow E$. Therefore, it is necessary to use high-dimensional probability symmetrization, i.e. to take the union of two edges, to eliminate this situation. The formula is as follows.

$$\text{edge}_{w_{ij}} = e^{\frac{-\text{dist}_{d_i, d_j} - \rho}{\vartheta}} \quad (3)$$

where, ϑ usually set to $\log_2 n$, n is the number of samples.

4.3. Spectral Embedding and Optimized Embedding

This section mainly introduces the steps of spectral and optimal embedding. Spectral embedding is a method that combines matrix decomposition with adjacency graphs. Adjacency graphs can easily represent low-dimensional initial coordinates. First, a simulated geometric adjacency graph is obtained, where each data point is a node, and edge weights between nodes represent their similarity in high-dimensional space. Then, a weighted adjacency matrix G is obtained, and a degree matrix W is obtained based on G, the Laplace matrix L is obtained according to the following formula, and L is used to decompose the eigenvalues. Finally, the initial low-dimensional coordinate P' is obtained by sorting of the feature values.

$$L = W^{\frac{1}{2}}(W - G)W^{\frac{1}{2}} \quad (4)$$

Algorithm 2 Optimize Embedding

```
Input: Spectral Embedding $P'$, n_epochs, min_dist, simple_set
Output: Optimize Dataset $P$
Initialize: $learning\_a = 1.0$
for each $e \leftarrow 1, \ldots, n\_epochs$ do
    for all ($[a, b]$, simple_set ) $\in top - \text{rep}_1$ do
        if Random ()$\leq$ simple_set then
            $p_a \leftarrow p_a +$ learning_$\alpha \cdot \nabla(\log m))\left(p_a, p_b\right)$
            for $i \leftarrow 1, \ldots, n-$ neg_samples do
                $c \leftarrow$ random sample from $P$
                $p_a \leftarrow p_a +$ learning_$\alpha \cdot \nabla(\log(1 - m))\left(p_a, p_c\right)$
            end for
        end if
    end for
     learning_$\alpha \leftarrow 1.0 - e/n$_epochs
end for
return  $P$
```

Algorithm 2 details the steps for optimizing the embedding. First, the weight of each data point is calculated to adjust its importance in the objective function by calculating the distance between each data point in the embedded space and its high-dimensional representation. The method used to calculate the weight is based on the neighborhood distance, which normalizes the distance between each data point and its neighbors, and maps the results to the range $[0, 1]$. The objective function is then minimized and the location of the points in the embedded space is updated using random gradient descent. Specifically, the difference between the distance between points in the embedded space and the similarity between data points in high-dimensional space is measured using an objective function based on cross-entropy (Song et al., 2022; de Boer et al., 2005; Lu and Steinerberger, 2022; Barthelme et al., 2020). Finally, the above steps are repeated until the objective function converges or until a predetermined number of iterations has been reached.

Optimized embedding aims at not changing the distance between data points in the low-dimensional spatial representation, but at finding a better representation of the low-dimensional manifold. This article uses the standard Euclidean distance (Malkauthekar, 2013; Gower, 1985) to represent the distance of the manifold. This must be passed in a parameter called min_dist (default=0.1) is used to control the degree to which points are closely grouped-defines the minimum distance between embedded points, the lower min_dist numerical value, the closer the embedded points are to each other and the resulting low-dimensional manifold structure will be more similar to the high-dimensional structure. After specifying a minimum distance, you can speed up the algorithm and reduce memory consumption by minimizing the cost coefficient using SGD (Arous et al., 2022; Barak et al., 2022; Wang and Joshi, 2021), as SGD can compute gradients from randomly sampled subsets, requiring only one subset to be retained. It is important to note that the gradient of the cost coefficient is a prerequisite for gradient descent. Since cross-entropy has the advantage of capturing the global structure of low-dimensional data, cross-entropy is used to represent the cost coefficient.

Algorithm 3 PM Algorithm

```
Input: tuple $a_i \in [-1, 1]$,    $a \in [-1, 1]^k$, privacy budget $\varepsilon$
Output: perturbed tuple $a' \in [-kc, ck]^k$
Initialize: $c = \frac{\mathrm{e}^{\varepsilon/2}+1}{\mathrm{e}^{\varepsilon/2}-1}$, Pro $= \frac{e^{\varepsilon}-e^{\frac{\varepsilon}{2}}}{2e^{\frac{\varepsilon}{2}}+2}$, $L\left(a_i\right) = -\frac{c+1}{2} \cdot a_i - \frac{c-1}{2}$, $H\left(a_i\right) = -L\left(a_i\right) + c - 1$, $\mathrm{a}' \leftarrow < 0, \dots, 0 >$
$t \leftarrow -\max\left\{1, \min\left\{k, \left|\frac{\varepsilon}{2.5}\right|\right\}\right\}$
Sample $t$ values uniformly without replacement from $\{1, 2, \dots, k\}$
for all$(i) \in [1, k]) \in top - \mathrm{rep}_1$ do
    Sample value $\upsilon$ uniformly at random from $[0, 1]$
    if $\upsilon < \frac{e^{\varepsilon/2}}{e^{\varepsilon/2}+1}$ then
        Sample $a_i'$ random from $\left[L\left(a_i\right), H\left(a_i\right)\right]$
        $c \leftarrow$ random sample from $P$
        $p_a \leftarrow p_a +$ learning_$\alpha \cdot \nabla(\log(1-m))\left(p_a, p_c\right)$
    else
        Sample $a_i'$ random from $\left(\left[-c, L\left(a_i\right)\right]\right) \cup \left(\left[H\left(a_i\right), c\right]\right)$
    end if
end for
$a' \leftarrow a_i'$
return $P$
```

4.4. LDP protection

This section mainly introduces the steps of LDP. After the acquisition of the data after dimensionality reduction, if we directly analyze the data, there is a risk of disclosure of private information. Therefore, we need to provide further LDP protection for P. An array P containing the coordinates of each data point in the specified low-dimensional space is obtained after optimizing and embedding the data, which can be thought of as a multi-dimensional array of numerical data. We cannot use LDP coding techniques directly because it is different from category data. To achieve our goal, we have adopted a segmentation mechanism for designing high-dimensional values, known as PM (Wang et al., 2019), to make LDP-processed data more effective.

Algorithm 3 shows the PM for high-dimensional numerical data. For t-dimensional data, the data still has high utility after LDP, where each attribute equally shares a privacy budget of ε. Specifically, the input high-dimensional tuple perturbs the data in each sampling dimension, defines an output domain $[-c, c]$ that is wider than the input domain $[-1, 1]$, and takes a random sample from the input domain. If the sample value $\upsilon < \frac{e^{\varepsilon/2}}{e^{\varepsilon/2}+1}$, then the perturbation value a_i' of a_i is randomly and uniformly sampled in $\left[L\left(a_i\right), H\left(a_i\right)\right]$. Otherwise, randomly and uniformly sample in $\left[-c, L\left(a_i\right)\right] \cup \left[H\left(a_i\right), c\right]$. Finally, the perturbation data of each sampling dimension are inserted into a' to obtain the perturbation data P^*.

4.5. Time complexity analysis

After dimensionality reduction of high-dimensional data, where $k \ll d$, the running time required for the PM algorithm is $O\left(\frac{m\sqrt{d\log d}}{\varepsilon}\right)$ (Wang et al., 2019), DPPro satisfies CDP when $k_1 > 2\left(\ln d + \ln\frac{2}{\delta}\right)$ (Xu et al., 2017), where DPPro achieves differential privacy by adding Gaussian noise, with a privacy parameter of $\delta = 10^{-5}$. The UMAP_PM algorithm uses uniform manifold approximation and projection for dimensionality reduction and adjusts the correlation between high-dimensional data using n_neighbors and min_dist to form a complex manifold structure and obtain a densely connected adjacency graph. The parameter n_neighbors determines the number of adjacent points, where a larger number of adjacent points indicates greater complexity among high-dimensional data attributes. The parameter min_dist represents the minimum effective distance between embedded points, and a lower value will result in a denser embedding. Therefore, as long as n_neighbors and min_dist are adjusted appropriately, and UMAP_PM is able to achieve a dimensionality reduction of $k < k_1$, it can effectively reduce the running time.

4.6. Privacy analysis

According to the definition of local differential privacy, for the above high-dimensional data tuple $a_i \in [-1, 1]^d$ and any low-dimensional perturbed data a_i' collected by a third-party server, to prove that the data processed by the algorithm in this paper satisfies $\varepsilon - LDP$, we only need to prove the following equation.

$$\frac{\Pr\left(M\left(a_i\right) = s^*\right)}{\Pr\left(M\left(a_i{}'\right) = s^*\right)} \leq e^{\varepsilon} \tag{5}$$

Since the PM algorithm belongs to a segmentation mechanism and satisfies the following formula:

$$\mathrm{Pf}\left(a_i{}' = x \mid a_i\right) = \begin{cases} \text{Pro}, & \text{if } x \in [L, H] \\ \frac{\text{Pro}}{e^{\varepsilon}}, & \text{if } x \in [-c, L] \cup [H, c] \end{cases} \tag{6}$$

where Pro is the high probability value for piecewise functions, $\frac{\mathrm{Pro}}{e^{\varepsilon}}$ is the low probability value for piecewise functions, $Pro = \frac{e^{\varepsilon}-e^{\frac{\varepsilon}{2}}}{2e^{\frac{\varepsilon}{2}}+2}$, $L\left(a_i\right)$ and $H\left(a_i\right)$ are mainly used to calculate where the high probability values of the piecewise functions start and end. Where $L\left(a_i\right) = -\frac{c+1}{2} \cdot a_i - \frac{c-1}{2}$, $H\left(a_i\right) = -L\left(a_i\right) + c - 1$.

Because the DP requirement of the subsection formula is that the upper bound of the ratio of the corresponding area of the high part of the probability density function to the corresponding area of the low part of the probability density function is e^{ε}, The ratio of the corresponding area of the part of the value with a high probability to the corresponding area of the part of the value with a low probability in this article is $\frac{(H-L)\,\mathrm{Pro}}{(2C-H-L)\cdot\frac{Pro}{e^{\varepsilon}}}$, Since $L\left(a_i\right)$ and $H\left(a_i\right)$ are given in Wang et al. (2019), it can be deduced from mathematical formulae that they always have the following formula that satisfies $\varepsilon - LDP$.

$$\frac{(H-L)\,\mathrm{Pro}}{(2c-H-L)\cdot\frac{\mathrm{Pro}}{e^{\varepsilon}}} \leq \frac{(c-1)e^{\varepsilon}}{2c+2-a_i} \leq e^{\varepsilon} \tag{7}$$

We use the UMAP_PM algorithm to decompose high-dimensional datasets. First, each sample in the high-dimensional data set is calculated with its local fuzzy simple set, and all local fuzzy simple sets are concatenated to obtain a high-dimensional adjacency topology representation weighting matrix. Then, to ensure that the final data match the correlation of the original high-dimensional data as closely as possible, spectral embedding and SGD algorithm are used for optimization embedding. Therefore, the weighting matrix and the final optimization matrix are only related to the parameter settings of the previous algorithm and will not reveal the privacy of the data.

When we interfere with two attributes using Alg.3, each attribute shares the privacy budget of $\frac{\varepsilon}{d}$ equally, and after dimensionality reduction, the data dimension is much smaller than d. Therefore, the privacy budget required for LDP in the dataset is increased. This can significantly improve the effectiveness of the data. Since the high dimensional data tuple a_i satisfies the formula (5), the reduced dimensional data tuple x satisfies the following formula:

$$\frac{\Pr\left(M(x) = s^*\right)}{\Pr\left(M\left(x'\right) = s^*\right)} \leq e^{\varepsilon} \tag{8}$$

In summary, our proposed UMAP_PM meets $\varepsilon - LDP$.

5. Evaluation

In this section, simulation experiments on real datasets are conducted for the proposed mechanism to evaluate and analyze the running time of synthetic datasets, the multidimensional probability distribution of synthetic datasets, and the accuracy of classification tasks on synthetic datasets.

Table 1
The Details of Datasets

Datasets	Type	Record (N)	Dimension (d)	Domain
SS13ACS	Integer	68725	30	2^{30}
Adult	Integer	45222	15	2^{15}
Criditcard	Integer	30000	24	2^{24}

Table 2
The Experimental Configuration Parameters

Datasets	n_neighbor	learning_rate	min_dist	n_epochs
SS13ACS	15	1.0	0.1	200
Adult	10	0.5	0.5	200
Criditcard	15	1.0	0.1	500

5.1. Datasets and Experimental Parameters

5.1.1. Datasets

In the simulation experiment, this paper uses three real datasets as shown in Table 1: (1) SS13ACS [3]: it is the American Community Survey (ACS) data, which captures the demographic, social, and economic characteristics of people living in the United States. (2) Adult [4]: it is a dataset from a classic data mining project, extracted from the 1994 US Census database and therefore known as the "Census Income" dataset, containing a total of 47522 records. (3) Creditcard [5]: it comes from the Creditcard customer dataset in Taiwan, the records include 24 dimensions, such as transaction, transaction type, default state, and so on, with a total of 30,000 pieces of information data. For the sake of simplicity, the attribute value ranges of the non-binary datasets are merged and compressed in the experiment.

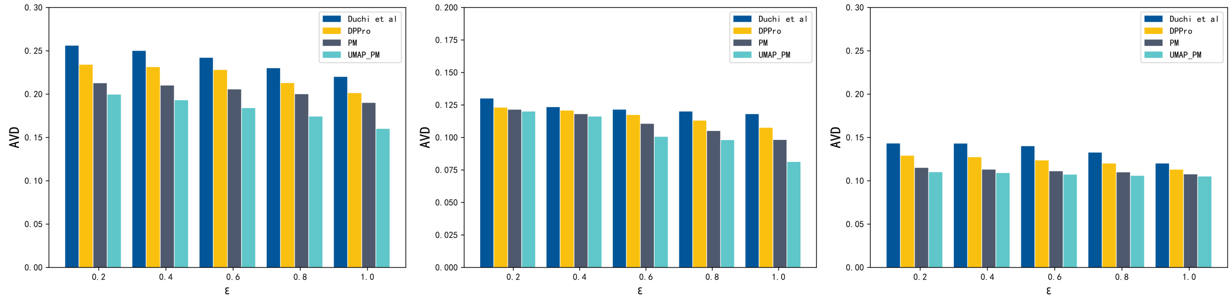
5.1.2. Experimental Parameters

All experiments were conducted using Python 3.9 as the programming language, Intel i5-8250 kernel as the hardware configuration for the experiment, using a 1.8GHz CPU frequency, 8GB of memory, and the Windows 10 operating system. In the process of synthesizing and releasing data, we use local differential privacy to process low-dimensional data privacy, form perturbed low-dimensional data tuples and send them to the server. The privacy parameters of all datasets during local differential privacy are $\varepsilon \in [0.2, 1.0]$. The optimal parameters for the three different datasets are as follows Table l. The n_neighbors determines the number of neighboring points, where a larger number indicates a higher number of neighbors. The min_dist is the effective minimum distance between embedding points, with lower values resulting in denser embedding. Both values are used to form a complex manifold structure in the high-dimensional dataset and to obtain a closely related adjacency graph. The experimental configuration parameters are given in Table 2.

[3] http://www.census.gov/acs/www/

[4] http://archive.ics.uci.edu/ml

[5] https://www.kaggle.com/datasets/uciml/default-of-credit-card-clients-dataset



(a) (b) (c)

Figure 2: The k-dimensional Statistical Query Accuracy (AVD vs. ε). (a) SS13ACS. (b) Adult. (c) Crditcard.

5.1.3. *Evaluating Indicator*

The purpose of the experiment is to evaluate the utility of the proposed UMAP_PM algorithm for data synthesis and whether the required run time for evaluation of synthesized data is reduced, and to verify whether this mechanism improves data utility while reducing run time. This includes four main aspects: First, statistical query accuracy. The average variation distance (AVD) is used to measure the average change distance between the synthesized and the original dataset. To evaluate the performance of UMAP_PM, a query set Qu containing 10000 random linear queries is generated for each dataset. Second, this article uses different dimensions on the SS13ACS and Creditcard datasets, and tests the AVD of the synthesized and original dataset using different numbers of users on the Adult and Creditcard, in order to test the effect of attribute dimensions and number of users on this algorithm in high-dimensional datasets. Third, for the purpose of testing the effectiveness of the proposed UMAP_PM algorithm, two methods, SVM classification and Logistic regression classification, have been applied for data analyses. Fourth, to compare and test the time of data synthesis with existing algorithms, and to verify whether the UMAP_PM algorithm can reduce the running time.

5.2. Evaluation results and analysis

5.2.1. *Statistical Query Accuracy*

In this section, we use the Average Variation Distance (AVD) to measure the difference between the joint probability distribution of the synthetic and original datasets. The larger the AVD, the greater the error between the estimated joint distribution $P(w)$ and the original joint distribution $Q(w)$. The calculation formula of AVD is as follows:

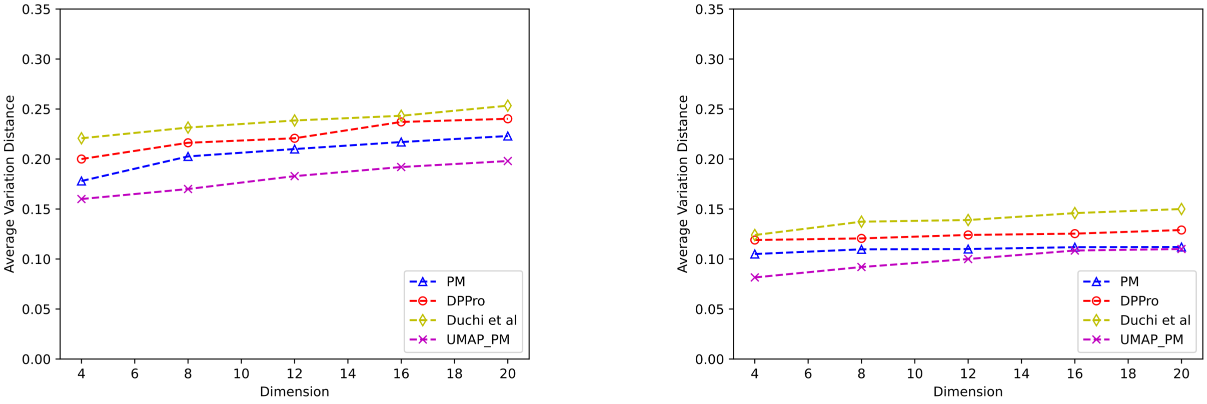
$$AVD(P,Q) = \frac{1}{2}\sum_{w\in\Omega} |P(w) - Q(w)| \tag{9}$$

As shown in Fig.2, the experiment was carried out on three datasets, SS13ACS, Adult and Creditcard. It can be seen that the accuracy of UMAP_PM is significantly better than that of the previous methods, in particular the DPPro algorithm. The reason can be attributed to the fact that it uses a popular structure to reduce the size of the data, and sets the number of neighboring points and the effective minimum distance between embedded points as reasonable settings to get a connected neighboring graph to reduce the size of the data, retaining as much of the underlying information of the original data as possible. As the privacy budget increases, the AVD gradually decreases, which is a reflection of the trade-off between privacy and utility.

5.2.2. *The Impact of Dimensions and User Numbers on the composite dataset*

In this section, we aim to investigate how different data dimensions and number of users affect the performance and quality of results achieved by the UMAP_PM algorithm on the synthetic dataset. We compared the average change distance of PM, Duchi et al.'s solution, DPPro, and UMAP_PM under the same privacy budget.

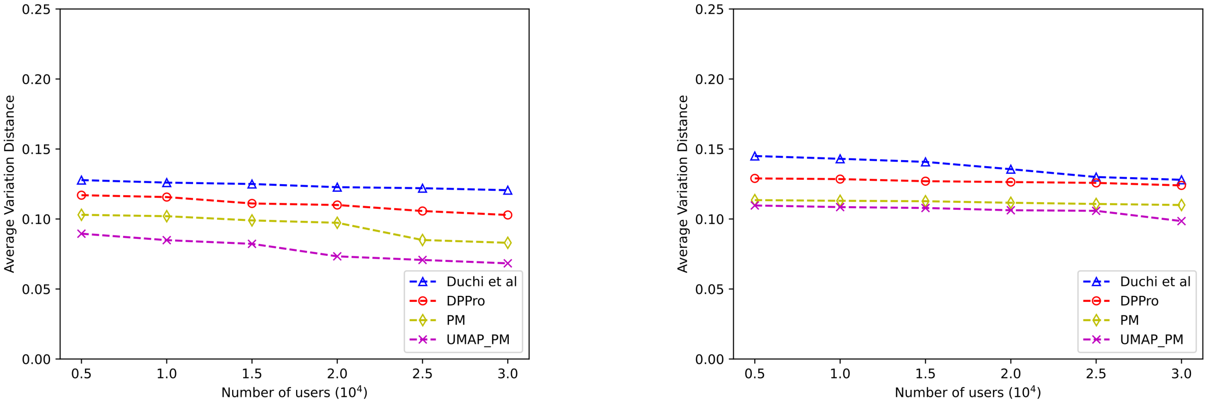
(1) AVD vs. d: In order to make the experimental results more convincing, we set the dimensions of the dimension experiment in the SS13ACS dataset and the Creditcard dataset, as the dimensions of SS13ACS and Creditcard are 24 and 30, respectively. In Fig.3, we describe the average change distance between the previous algorithm and UMAP_PM under the same privacy budget ε. It can be seen that UMAP_PM is not affected by the dimension and its performance is stable, while the average change distance gradually increases and the effectiveness of the previous algorithm becomes significantly worse as the dimension is continuously increased. It can be explained by the fact that the more dimensional the data, the more complex the interdependencies and the more correlated the data. When the dimension is fixed, the average change distance of the UMAP_PM algorithm is always lower than that of the previous algorithm, reflecting that the UMAP_PM algorithm captures only low-dimension data, reduces the perturbation error introduced after local differential privacy processing, and provides a higher data benefit. When the privacy parameters are fixed, the average change distance gradually increases with increasing d. When the dimension d is fixed unchanged, the average change distance of the UMAP_PM algorithm is always lower than that of the previous algorithm. As a result, the UMAP_PM algorithm achieves higher data utility compared to other algorithms.



(a)

(b)

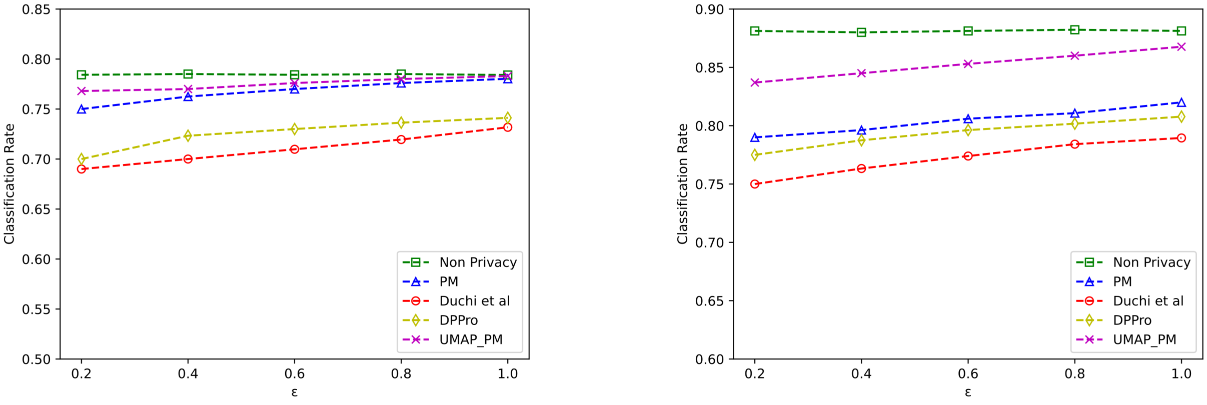
Figure 3: AVD vs. Dimensions (AVD vs. d). (a) SS13ACS. (b) Crditcard.



(a)

(b)

Figure 4: AVD vs. Number of Users (AVD vs. N). (a) Adult. (b) Crditcard.



(a)

(b)

Figure 5: The Classification Utility of SS13ACS Dataset. (a) SVM. (b) Logistic Regression.

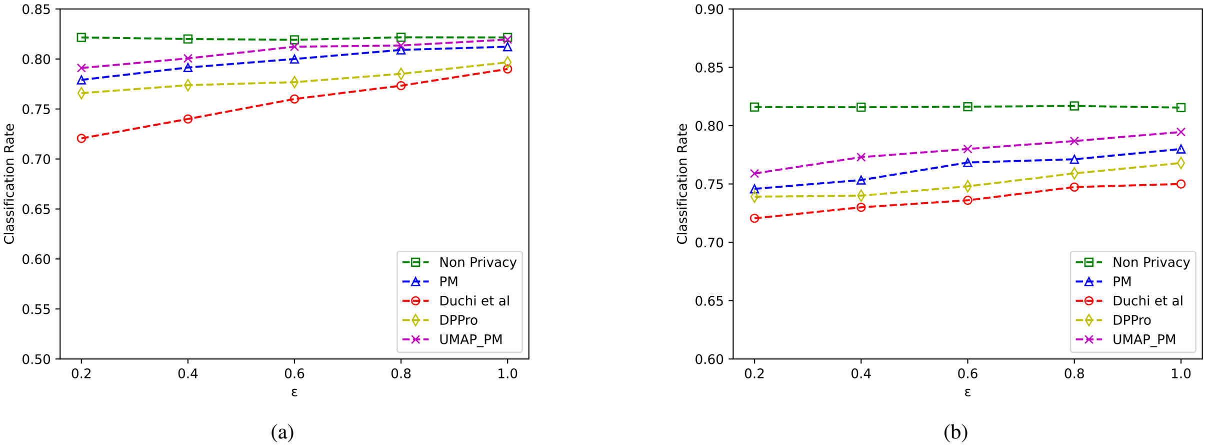


Figure 6: The Classification Utility of Adult Dataset. (a) SVM. (b) Logistic Regression.

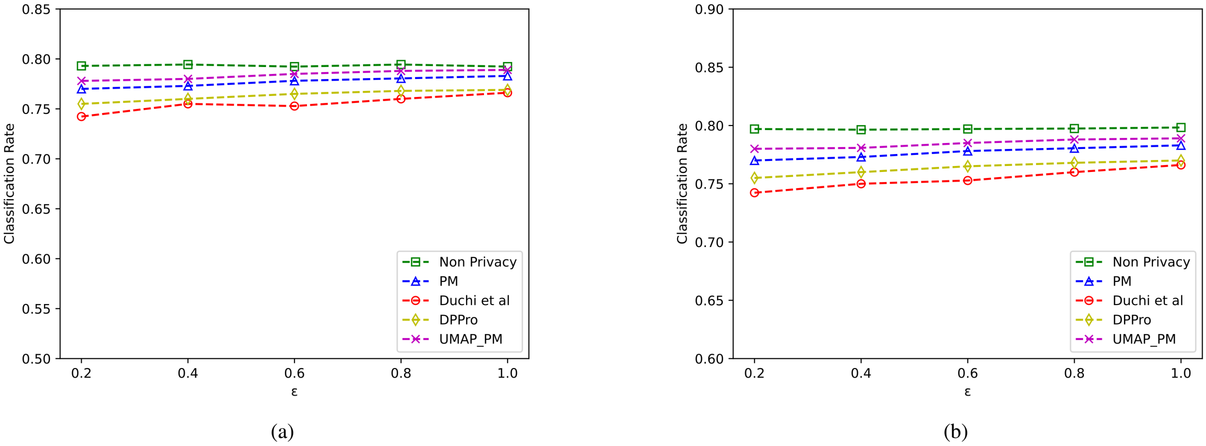


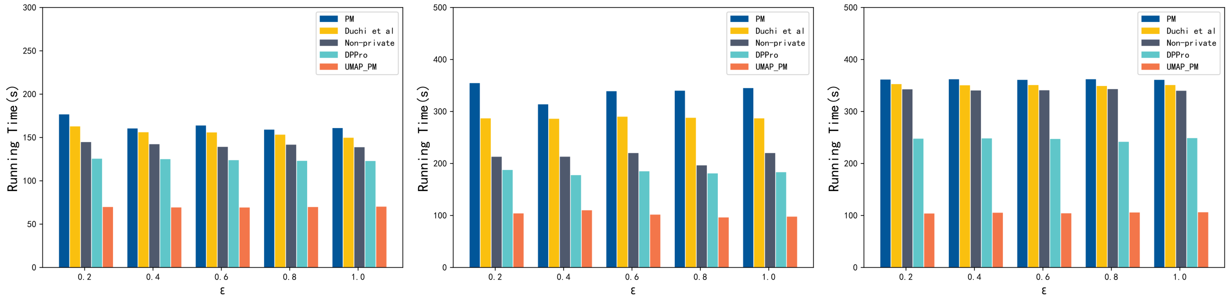
Figure 7: The Classification Utility of Creditcard Dataset. (a) SVM. (b) Logistic Regression.

(2) AVD vs. N: This experiment is compared with the most advanced PM, the solution of Duchi et al., and the DPPro algorithm, to test the impact of the number of users on the synthetic dataset of the UMAP_PM algorithm under the same dimension and the same privacy conditions. We use two datasets, Adult and Creditcard, with smaller dimensions to test the changing trend of AVD with the number of users to reduce the running time of the algorithm. As shown in Fig.4, the AVD of the three datasets gradually decreases as the number of users increases, when the data dimension and privacy parameters are fixed. This is because the average change distance between the synthetic dataset and the original dataset becomes smaller as the number of users increases, and the training results become more accurate. The AVD of the UMAP_PM algorithm is significantly lower than that of the previous algorithms, which is a good indication of its usefulness. Overall, the more users, the lower the dimensionality and the more accurate the results. The above two experiments show that the synthetic dataset of the UMAP_PM algorithm has a high degree of utility.

5.2.3. The Classification Utility

In this section, we evaluate the accuracy performance of the proposed methods for logistic regression and SVM classification. We use the numerical attributes 'birth', 'total income', and 'late payment next month' as dependent variables and all other attributes as independent variables for the SS13ACS, Adult, and Creditcard datasets. We encode and convert each classification attribute with a value into an integer from $[-1, 0]$, following common practice, using the scikit-learn library. We use 80% of the records as the training set and 20% of the records as the test set in each dataset, using 10-fold cross-validation 20 times to evaluate the performance of each method on each dataset to ensure the accuracy of the evaluation.

Fig.5, 6, 7 describe the classification accuracy of SVM and Logistic regression respectively with changing privacy



(a)

(b)

(c)

Figure 8: Running Time. (a) SS13ACS. (b) Adult. (c) Crditcard.

parameter ε on three datasets. It can be seen that the classification accuracy of SVM and logistic regression shows an upward trend with ε increases. This indicates that privacy has a certain impact on the utility of the data. When ε is small, the direct privacy protection of the original high-dimensional data by using the methods of PM and Duchi et al. This will lead to the introduction of a large amount of noise. The UMAP_PM algorithm first reduces the dimensionality and then perturbs the low-dimensional data, which reduces the perturbation errors to some extent, resulting in a significant improvement in the utility of synthesizing high-dimensional datasets. UMAP_PM uses graph theory to describe the similarity between data points, maintaining the dependency relationships between high-dimensional data attributes by constructing adjacency graphs. In contrast, the DPPro algorithm uses random projection for dimensionality reduction, and the random selection of projection matrices makes the synthesized data uncertain and destroys the complex dependency relationships between different attributes in high-dimensional data sets. This makes it difficult to effectively capture the non-linear structure in the original high-dimensional data, resulting in a significant decrease in the utility of the data.

5.2.4. Running Time

In this section, the runtime of high-dimensional data release is experimentally evaluated. Fig.8 shows the running time for the release of new records in SS13ACS, Adult and Creditcard. The algorithm proposed in this paper has a significantly lower running time than previous algorithms for releasing new datasets. This is because we use UMAP_PM to reduce the dimension of high-dimensional data. Since this method has the characteristics of no calculation limit on the embedded dimension, faster execution speed and smaller memory occupancy, it reduces the time complexity of the algorithm and reduces the execution time. In addition, the credit card and SS13ACS datasets take significantly less time to process compared to an Adult dataset, because UMAP_PM has better repeatability. The larger the dataset, the more obvious the advantage in terms of reduced running time. Overall, the run time for the release of high dimensional data is significantly reduced with UMAP_PM.

When the cross-attention module is removed, when $\epsilon = 100$, the mean value of the accuracy difference between DP-MCDA and DP-MCDA (w/o CA) in the three domains of the Office-31 dataset is about 3.8%, and the mean value of the accuracy difference in the two domains of DomainNet is about 13.8%. The reason for such a large difference in the average value is that the three domains of the Office-31 dataset are relatively not very different in distribution, all of them are color images of office supplies; while the DomainNet domains have huge differences in distribution, for example, the graffiti in the *DQ* domain has almost no similarity with the physical image in the *DR* domain, and it is difficult to correspond the two even with human associative ability, so after removing the cross-attention module, the accuracy in both tasks of the DomainNet dataset declines significantly.

In summary, both the differential privacy module and the cross-attention module are essential for DP-MCDA. The differential privacy module provides privacy protection; the cross-attention module assigns different weights to each source domain, which reduces domain bias between source domains and allows the model to better migrate knowledge from source domains to the target domain. Overall, both improve the accuracy of image classification while protecting sensitive data in multi-source domain adaptation training, achieving a balance between privacy and the utility of the differential privacy mechanism in multi-source domain adaptation.

6. Conclusion

In this paper, we propose a novel solution, UMAP_PM, to achieve high-dimensional data release with LDP. Specifically, UMAP_PM builds the correlation of attributes, synthesizing an approximate dataset for privacy protection. Our method effectively reduces the running time and minimizes the amount of added noise learned from high-dimensional data records. Therefore, this method simultaneously guarantees the practicality and privacy of the released data. The experiment results on the real datasets show that UMAP_PM is an efficient and effective mechanism to release a high-dimensional dataset while providing sufficient local differential privacy guarantee for users. During the dimensionality reduction process, the parameters are known from experience or set manually, and the choice of parameters affects the mapping results, the speed of dimensionality reduction and the accuracy of the low-dimensional space. Smaller parameters may improve the speed of computation but have a negative impact on the accuracy of the dimensionality reduction. Larger parameters may improve the accuracy of dimensionality reduction, but reduce the speed of computation. Therefore, how to use estimation methods such as eigenvalue or mapping methods, geometric learning methods, and adaptive parameter determination required in manifold learning algorithms using statistical learning methods is an urgent problem for the future.

CRediT authorship contribution statement

Aixin Lin: Conceptualization, Methodology, Software, Validation, Writing - Original Draft. **Xuebin Ma:** Conceptualization, Writing - Review Editing, Supervision.

Acknowledgements

This work was supported by the Self-project Program of Engineering Research Center of Ecological Big Data, Ministry of Education; and the University Youth Science and Technology Talent Development Project (Innovation Group Development Plan) of Inner Mongolia A. R. of China [grant number NMGIRT2318].

References

Y. Zhong, P. Chalise, J. He, Nested cross-validation with ensemble feature selection and classification model for high-dimensional biological data, Commun. Stat. Simul. Comput. 52 (2023) 110–125.

H. Zhao, S. You, C. Zhang, Research on the evaluation of economic vitality of districts in beijing on experimental and mathematical statistics analysis, in: International Conference on Statistics, Data Science, and Computational Intelligence (CSDSCI 2022), volume 12510, SPIE, 2023, pp. 126–137.

M. F. Kabir, T. Chen, S. A. Ludwig, A performance analysis of dimensionality reduction algorithms in machine learning models for cancer prediction, Healthcare Analytics 3 (2023) 100125.

X. Xiong, S. Liu, D. Li, Z. Cai, X. Niu, A comprehensive survey on local differential privacy, Secur. Commun. Networks 2020 (2020) 8829523:1–8829523:29.

R. Chang, Y. Chang, C. Wang, Outsourced k-means clustering for high-dimensional data analysis based on homomorphic encryption, J. Inf. Sci. Eng. 39 (2023) 525–548.

C. Dwork, F. McSherry, K. Nissim, A. D. Smith, Calibrating noise to sensitivity in private data analysis, in: S. Halevi, T. Rabin (Eds.), Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006, Proceedings, volume 3876 of *Lecture Notes in Computer Science*, Springer, 2006, pp. 265–284. URL: https://doi.org/10.1007/11681878_14. doi:10.1007/11681878_14.

T. Wang, W. Yang, X. Ma, B. Wang, Event-set differential privacy for fine-grained data privacy protection, Neurocomputing 515 (2023) 48–58.

R. Redberg, Y. Zhu, Y.-X. Wang, Generalized ptr: User-friendly recipes for data-adaptive algorithms with differential privacy, in: F. Ruiz, J. Dy, J.-W. van de Meent (Eds.), Proceedings of The 26th International Conference on Artificial Intelligence and Statistics, volume 206 of *Proceedings of Machine Learning Research*, PMLR, 2023, pp. 3977–4005. URL: https://proceedings.mlr.press/v206/redberg23a.html.

Z. Zhang, X. Xu, F. Xiao, LGAN-DP: A novel differential private publication mechanism of trajectory data, Future Gener. Comput. Syst. 141 (2023) 692–703.

W. Zhang, J. Zhao, F. Wei, Y. Chen, Differentially private high-dimensional data publication via markov network, EAI Endorsed Trans. Security Safety 6 (2019) e4.

K. Wu, P. Chen, O. Ghattas, A fast and scalable computational framework for large-scale high-dimensional bayesian optimal experimental design, SIAM/ASA Journal on Uncertainty Quantification 11 (2023) 235–261.

X. Lu, C. Piao, J. Han, Differential privacy high-dimensional data publishing method based on bayesian network, in: 2022 International Conference on Computer Engineering and Artificial Intelligence (ICCEAI), 2022, pp. 623–627. doi:10.1109/ICCEAI55464.2022.00132.

J. Zhao, T. Gao, J. Zhang, Method of local differential privacy method for high-dimensional data based on improved bayesian network (in chinese), Netinfo Security 23 (2023).

J. Zhang, G. Cormode, C. M. Procopiuc, D. Srivastava, X. Xiao, Privbayes: Private data release via bayesian networks, ACM Trans. Database Syst. 42 (2017) 25:1–25:41.

N. Wang, X. Xiao, Y. Yang, T. D. Hoang, H. Shin, J. Shin, G. Yu, Privtrie: Effective frequent term discovery under local differential privacy, in: 34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018, IEEE Computer Society, 2018, pp. 821–832. URL: https://doi.org/10.1109/ICDE.2018.00079. doi:10.1109/ICDE.2018.00079.

X. Zhang, L. Chen, K. Jin, X. Meng, Private high-dimensional data publication with junction tree (in chinese), J. Comput. Res. Dev 55 (2018) 2794–2809.

X. Yang, T. Wang, X. Ren, W. Yu, Copula-based multi-dimensional crowdsourced data synthesis and release with local privacy, in: 2017 IEEE Global Communications Conference, GLOBECOM 2017, Singapore, December 4-8, 2017, IEEE, 2017, pp. 1–6. URL: https://doi.org/10.1109/GLOCOM.2017.8253989. doi:10.1109/GLOCOM.2017.8253989.

T. Wang, X. Yang, X. Ren, W. Yu, S. Yang, Locally private high-dimensional crowdsourced data release based on copula functions, IEEE Trans. Serv. Comput. 15 (2022) 778–792.

Q. Xue, Y. Zhu, J. Wang, Joint distribution estimation and naïve bayes classification under local differential privacy, IEEE Trans. Emerg. Top. Comput. 9 (2021) 2053–2063.

F. Wei, W. Zhang, Y. Chen, J. Zhao, Differentially private high-dimensional data publication via markov network, in: R. Beyah, B. Chang, Y. Li, S. Zhu (Eds.), Security and Privacy in Communication Networks - 14th International Conference, SecureComm 2018, Singapore, August 8-10, 2018, Proceedings, Part I, volume 254 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, Springer, 2018, pp. 133–148. URL: https://doi.org/10.1007/978-3-030-01701-9_8. doi:10.1007/978-3-030-01701-9_8.

D. Wang, J. Xu, Principal component analysis in the local differential privacy model, Theor. Comput. Sci. 809 (2020) 296–312.

J. Ge, Z. Wang, M. Wang, H. Liu, Minimax-optimal privacy-preserving sparse PCA in distributed systems, in: A. J. Storkey, F. Pérez-Cruz (Eds.), International Conference on Artificial Intelligence and Statistics, AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain, volume 84 of *Proceedings of Machine Learning Research*, PMLR, 2018, pp. 1589–1598. URL: http://proceedings.mlr.press/v84/ge18a.html.

C. Xu, J. Ren, Y. Zhang, Z. Qin, K. Ren, Dppro: Differentially private high-dimensional data release via random projection, IEEE Trans. Inf. Forensics Secur. 12 (2017) 3081–3093.

Z. Gu, G. Zhang, C. Ma, L. Song, Differential privacy data publishing method based on the probabilistic principal component analysis (in chinese), Journal of Harbin Engineering University 42 (2021) 1217–1223.

H. Sun, J. Yang, X. Cheng, et al., A high-dimensional numeric data collection algorithm for local difference privacy based on random projection (in chinese), Big Data Res 6 (2020) 3–11.

X. Li, C. Luo, P. Liu, L.-e. Wang, D. Yu, Injecting differential privacy in rules extraction of rough set, in: Proceedings of the 2nd International Conference on Healthcare Science and Engineering 2nd, Springer, 2019, pp. 175–187.

Z. Qin, Y. Yang, T. Yu, I. Khalil, X. Xiao, K. Ren, Heavy hitter estimation over set-valued data with local differential privacy, in: E. R. Weippl, S. Katzenbeisser, C. Kruegel, A. C. Myers, S. Halevi (Eds.), Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016, ACM, 2016, pp. 192–203. URL: https://doi.org/10.1145/2976749.2978409. doi:10.1145/2976749.2978409.

G. Fanti, V. Pihur, Ú. Erlingsson, Building a RAPPOR with the unknown: Privacy-preserving learning of associations and data dictionaries, Proc. Priv. Enhancing Technol. 2016 (2016) 41–61.

M. Xu, B. Ding, T. Wang, J. Zhou, Collecting and analyzing data jointly from multiple services under local differential privacy, Proc. VLDB Endow. 13 (2020) 2760–2772.

N. Wang, X. Xiao, Y. Yang, J. Zhao, S. C. Hui, H. Shin, J. Shin, G. Yu, Collecting and analyzing multidimensional data with local differential privacy, in: 35th IEEE International Conference on Data Engineering, ICDE 2019, Macao, China, April 8-11, 2019, IEEE, 2019, pp. 638–649. URL: https://doi.org/10.1109/ICDE.2019.00063. doi:10.1109/ICDE.2019.00063.

Ú. Erlingsson, V. Pihur, A. Korolova, RAPPOR: randomized aggregatable privacy-preserving ordinal response, in: G. Ahn, M. Yung, N. Li (Eds.), Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, November 3-7, 2014, ACM, 2014, pp. 1054–1067. URL: https://doi.org/10.1145/2660267.2660348. doi:10.1145/2660267.2660348.

X. Zhang, N. Fu, X. Meng, Towards spatial range queries under local differential privacy (in chinese), J. Comput. Res. Dev. 57 (2020) 847.

S. Kim, H. Shin, C. H. Baek, S. Kim, J. Shin, Learning new words from keystroke data with local differential privacy, IEEE Trans. Knowl. Data Eng. 32 (2020) 479–491.

T. Wang, J. Blocki, N. Li, S. Jha, Locally differentially private protocols for frequency estimation, in: E. Kirda, T. Ristenpart (Eds.), 26th USENIX Security Symposium, USENIX Security 2017, Vancouver, BC, Canada, August 16-18, 2017, USENIX Association, 2017, pp. 729–745. URL: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/wang-tianhao.

Z. Li, T. Wang, M. Lopuhaä-Zwakenberg, N. Li, B. Skoric, Estimating numerical distributions under local differential privacy, in: D. Maier, R. Pottinger, A. Doan, W. Tan, A. Alawini, H. Q. Ngo (Eds.), Proceedings of the 2020 International Conference on Management of Data, SIGMOD Conference 2020, online conference [Portland, OR, USA], June 14-19, 2020, ACM, 2020, pp. 621–635. URL: https://doi.org/10.1145/3318464.3389700. doi:10.1145/3318464.3389700.

W. Song, C. Zheng, C. Huang, L. Liu, Heuristically mining the top-k high-utility itemsets with cross-entropy optimization, Appl. Intell. 52 (2022) 17026–17041.

P. de Boer, D. P. Kroese, S. Mannor, R. Y. Rubinstein, A tutorial on the cross-entropy method, Ann. Oper. Res. 134 (2005) 19–67.

J. Lu, S. Steinerberger, Neural collapse under cross-entropy loss, Applied and Computational Harmonic Analysis 59 (2022) 224–241.

A. Barthelme, R. Wiesmayr, W. Utschick, Model order selection in doa scenarios via cross-entropy based machine learning techniques, in: 2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020, IEEE, 2020, pp. 4622–4626. URL: https://doi.org/10.1109/ICASSP40776.2020.9053029. doi:10.1109/ICASSP40776.2020.9053029.

M. Malkauthekar, Analysis of euclidean distance and manhattan distance measure in face recognition, in: Third International Conference on Computational Intelligence and Information Technology (CIIT 2013), IET, 2013, pp. 503–507.

J. C. Gower, Properties of euclidean and non-euclidean distance matrices, Linear algebra and its applications 67 (1985) 81–97.

G. B. Arous, R. Gheissari, A. Jagannath, High-dimensional limit theorems for SGD: effective dynamics and critical scaling, in: NeurIPS, 2022. URL: http://papers.nips.cc/paper_files/paper/2022/hash/a224ff18cc99a71751aa2b79118604da-Abstract-Conference.html.

B. Barak, B. L. Edelman, S. Goel, S. M. Kakade, E. Malach, C. Zhang, Hidden progress in deep learning: SGD learns parities near the computational limit, in: NeurIPS, 2022. URL: http://papers.nips.cc/paper_files/paper/2022/hash/884baf65392170763b27c914087bde01-Abstract-Conference.html.

J. Wang, G. Joshi, Cooperative SGD: A unified framework for the design and analysis of local-update SGD algorithms, J. Mach. Learn. Res. 22 (2021) 213:1–213:50.

Aixin Lin is a master's student at the School of Computer Science, Inner Mongolia University. Her main research interests are privacy preservation and high-dimensional data, focusing on privacy preservation in local differential privacy, high-dimensional data and machine learning.

Xuebin Ma is an associate professor in the School of Computer Science at Inner Mongolia University. He mainly researches on big data analytics techniques, focusing on federated learning, privacy protection, and secure data sharing and utilization. He has also conducted research in areas such as wireless networks and delay-tolerant networks. In addition, he also conducts research on collaborative topics in the field of information security.

LaTeX 源文件

Declaration of Interest Statement

Declaration of interests

☐The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☒The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Xuebin Ma reports financial support was provided by Inner Mongolia Autonomous Region Department of Science and Technology.