Computer Science and Information Systems 16(3):705–731 https://doi.org/10.2298/CSIS180915023L

Research on Improved Privacy Publishing Algorithm Based on Set Cover

Haoze Lv[1], Zhaobin Liu[1], Zhonglian Hu[1], Lihai Nie[2], Weijiang Liu[1], and Xinfeng Ye[3]

[1] School of Information Science and Technology, Dalian Maritime University
China
Correspondence: zhbliu@dlmu.edu.cn
[2] Division of Intelligence and Computing, Tianjin University
China
nlh3392@tju.edu.cn
[3] Department of Computer Science, University of Auckland
New Zealand
x.ye@auckland.ac.nz

Abstract. With the invention of big data era, data releasing is becoming a hot topic in database community. Meanwhile, data privacy also raises the attention of users. As far as the privacy protection models that have been proposed, the differential privacy model is widely utilized because of its many advantages over other models. However, for the private releasing of multi-dimensional data sets, the existing algorithms are publishing data usually with low availability. The reason is that the noise in the released data is rapidly grown as the increasing of the dimensions. In view of this issue, we propose algorithms based on regular and irregular marginal tables of frequent item sets to protect privacy and promote availability. The main idea is to reduce the dimension of the data set, and to achieve differential privacy protection with Laplace noise. First, we propose a marginal table cover algorithm based on frequent items by considering the effectiveness of query cover combination, and then obtain a regular marginal table cover set with smaller size but higher data availability. Then, a differential privacy model with irregular marginal table is proposed in the application scenario with low data availability and high cover rate. Next, we obtain the approximate optimal marginal table cover algorithm by our analysis to get the query cover set which satisfies the multi-level query policy constraint. Thus, the balance between privacy protection and data availability is achieved. Finally, extensive experiments have been done on synthetic and real databases, demonstrating that the proposed method preforms better than state-of-the-art methods in most cases.

Keywords: Differential Privacy, Set Cover, Frequent Itemsets, Marginal Table.

1. Introduction

With the rising attention of big data, the amount of published information is rapidly increasing. Online office, cloud storage, data synchronization and other services have brought convenience for us, but also generate a huge user data aggregation. These data contain a wealth of valuable information, therefore, data releasing and sharing has become an important topic both in scientific research and information industry [1]. However, directly releasing original data can lead to information disclosure of user. So, how to publish

the data while protecting the user's personal privacy has become an important issue. In the scenario with privacy protection requirements, the differential privacy model is widely used because it doesn't need attack hypothesis and background knowledge of attackers, but it can quantify and analyze the privacy risk [2] [3].

However, when publishing the multi-dimensional data sets, current privacy models often have low performances in both privacy protection and data availability. The reasons are explained as follows. First, the noise added into each dimension in the released data will undoubtedly increase when the dimension of the data becomes larger. Second, the query results are usually not very valid due to large cumulative noise of data. In view of this issue, we propose a privacy-preserving data releasing algorithm based on regular marginal tables to reduce the noise and improve the data availability.

To figure out the issue of regular marginal table, we propose a publishing algorithm to reduce the dimension of the cover data set in the same dimension marginal table, thus achieving differential privacy protection with Laplace noise [4]. We first introduces how to choose the dimension k of marginal tables in a cover σ. Then we choose a smallest set of k-marginal tables satisfying the cover on the condition of a proper k. Different from the classic algorithms in this step, we use frequent item to measure the importance of the k-marginal tables. The situation that k-marginal tables with more frequent attributes are usually with higher data availability.

Meanwhile, we propose a differential privacy model with irregular marginal table partitioning and find the marginal table query cover set which satisfies the multi-level query policy constraint. This model is proposed to be used in application scenarios with low data privacy protection requirements and high cover requirements. Thus, the balance between privacy protection and data availability is achieved. The main contributions of this paper is summarized as follows.

1. We do many research on bounds of k-marginal table cover and propose to use frequent attributes to increase the usability of the released data.
2. We present a k-marginal tables publishing algorithm based on frequent items and regular marginal benefits, which is with lower time complexity and higher data usability.
3. We introduce improved marginal table differential privacy publishing algorithms based on irregular marginal table in this paper in detail.
4. We conduct extensive experiments in both public databases, demonstrating that the presented algorithm always performs better than the state-of-the-art approaches.

The rest of this paper is organized as follows. Section 2 shows the related work of this paper. Then we will present regular marginal table differential privacy publishing algorithm under frequent item set algorithms in Section 3. Another algorithm with some theorems and proofs we give in section 4 is used in the scenario of irregular marginal table. We make performance analysis by experiments in Section 5. Finally, Section 6 will draw some conclusions.

2. Related Work

2.1. Differential Privacy

In literature, variable kinds of privacy-preserving algorithms [5], [6], [7] have been proposed to prevent the sensitive information from being disclosed. Among these algorithms,

perhaps the most well-known algorithm is K-anonymous algorithm proposed by Sweeney in 2002 [8]. K-anonymous algorithm has the characteristic that a record can be protected with at least $k-1$ other records with respect to the quasi-identifier. However, K-anonymous algorithm is only focused on identity protection. To overcome the drawback, Machanavajjhala et al. [9] proposed L-diversity algorithm inspired by K-anonymous algorithm to provide sufficient protection against attribute disclosure. However, the premise of data generalization is to know the background knowledge of the attacker, which is always impossible. Thus the data privacy protection technology that encrypts the sensitive information is proposed.

Recently, the major efforts in this field are paid to reduce noise while protecting privacy at the same time, including histogram [10] and contingency tables [11]. But with the increasing popularization of the big data, the amount of user information exploded. Querying the publishing data is a problem with high sensitivity because of large quantity noise. Therefore the Flat Method [12], Matrix [13], Data Cube [14] do not work with big data.

To solve the problem, Dwork etc. proposed a differential privacy protection model [15] using the Laplace mechanism. It is a common mechanism for achieving differential privacy by adding Laplace noise to numerical data. Therefore, the method is mainly used in the privacy protection release process of statistical data. Laplace differential privacy model solves the problem that the attacker can easily find the user's privacy information through the normalized query strategy. Meanwhile, it can also achieve a high performance in big data.

The basic idea of the differential privacy model is to randomly perturb the published data, so that the attacker cannot obtain the private information of individual from the published data regardless of any background knowledge and any data mining and analysis information. The advantages of this model are that there is no need to make special assumptions about the attacker's background knowledge and the specific attack method. Thus, we use the privacy budget to analyze the risk of data disclosure quantitatively.

2.2. Set Cover Problem

Set Cover Problem (SCP) is a classical problem in combinatorial mathematics, computer science and computer complexity theory. This paper uses the idea of set cover to form mathematically model and improve the middle ware partitioning problem in the differential privacy model. Therefore, this part will introduce the basic concepts of set cover problems. Set cover problem is divided into two categories based on the definition of the problem: the determinant problem [16] and the optimization problem of set cover [17]. The goal of set cover optimization problem is to find cover sets that meet the optimization conditions.

SCP has been proved to be a NPC or NPH problem [18] as early as 1976. As a result, the study of the approximate algorithm of the optimal cover set is one of the important point of this issue which is an approximation optimal set selected from the candidate through the algorithm. The goal of the approximation algorithm is to reduce the time complexity of the algorithm by obtaining the approximate solution [19]. Thus, the approximation algorithm cares about two aspects of the problem: time complexity and approximate degree of solution.

As early 1970, Edmonds has proved that the greedy algorithm for linear functions must be optimal in the matroid structure [20]. However, the problem in practical applications is not linear but a submodule function. The submodule function is a formal description of the "marginal utility decrement" in the set theory. In recent years, in the research of the modulo algorithm of the submodule function, Sagnol G proposed the modulo algorithm of the submodule function in polynomial time [21], and applied it to the maximum cover problem. He proved that unless $P = NP$ is satisfied, the greedy algorithm is the best maximization algorithm in polynomial time for the condition that satisfies the submodule function. In order to solve the problem of finding the minimum submodule cover set, Pengjun Wan et al. proposed MSC/SC (Minimum Submodular Cover with Submodular Cost)[22]. The pseudo code of the algorithm is shown in Algorithm 1.

Algorithm 1: MSC

Input : a collection of E
Output: cover collection X

$X \leftarrow \emptyset$
while $\exists e \in E$ such that $\Delta_e f(X) > 0$ **do**
    select $x \in E$ with maximum $\Delta_e f(X)/c(x)$
    $X \leftarrow X \cup \{x\}$

Nevertheless, there may be a risk that the usability of the data is reduced by mixing too much noise based on contingency table. To overcome this problem, Qardaji et al. proposed a differential privacy released algorithm based on marginal table [23]. Noise is effectively reduced based on marginal table differential privacy model. However, relevance of the real data set of attributes is not taken into account. The marginal table contains a part of invalid query combination, which reduces the data availability. In view of this problem, we propose marginal table differential privacy publishing algorithms based on frequent item sets.

There are many candidate methods for mining frequent items [24], [25], [26], [27], [28] and we choose well-known Apriori [29] to analyze the data set in practical application, considering about the marginal table support and marginal benefit. Then the majority of valid query combinations is covered by generated marginal table and the data availability of query middle ware is improved further.

2.3. Differential Privacy Publishing Algorithm based on Marginal Table

Differential privacy is a model to protect information for data publishing. Therefore, in the scenario of multi-dimensional or high-dimensional data publishing, the research and application of differential privacy publishing algorithm is necessary. Wahbeh Qardaji et al proposed a differential privacy publishing algorithm called PriView [30] in 2014. The algorithm overcomes the problem of large added-noise and low data availability when publishing high-dimensional data sets. Its key idea is to cover the partial query range of high-dimensional data by using multiple marginal table cover sets composed of tables of the same dimension. This section will give a brief introduction to its improvement methods and related technologies.

Middle Ware of Publishing Marginal Table The marginal table is a new type of query middle ware derived from the contingency table proposed by Wahbeh Qardaji. It can be obtained by splitting, extracting, and recombining the attributes included in the contingency table. It's noise has been significantly reduced compared to the contingency table.

In the differential privacy publishing algorithm of the Laplace mechanism, the noise added in the analysis of the query middle ware can be reduced by publishing a low-dimensional marginal table. However, the shortcomings of the marginal table also exist. The marginal table implements dimension reduction by truncating the attributes of the contingency table, which certainly leads to a reduction in the scope of the query. However, in high-dimensional data sets, the noise error made by the contingency table publishing method will be too hard to estimated. What's more, as the data dimension becomes higher, the contingency table may become more and more sparse, which means the result of multi-attribute queries will be 0 or lack of realistic query meaning. Therefore, we sacrifice some of the meaningless high-dimensional queries and use the marginal table instead of the contingency table publishing in the differential privacy publishing algorithm. It has certain practical application value in reducing noise and improving data availability.

Laplace Noise Error Analysis under Marginal Table The marginal differential privacy publishing algorithm publish a cover set of the original d-dimensional data set consisting of n k-dimensional marginal tables. At present, the noise mixing method used for the marginal table cover set is mainly the direct method. Since the middle ware is a combination of multiple tables, in order to facilitate the noise error analysis, Wahbeh Qardaji proposed the ESE (Expected Squared Error) noise estimation method. This method estimates the noise error of the single marginal table by adding the each noise-added result m_{ij} from each item to the variance of the real result n_{ij}. The calculation formula is as shown in Eq. 1.

$$ESE = \sum_{j=1}^{2^k} (m_{ij} - n_{ij})^2 \tag{1}$$

The direct method is proposed by Dwork et al [31]. This method directly adds the noise of $Laplace(n/\varepsilon)$ to each item in the same-dimensional marginal table. It proves that the marginal table cover set proposed by the method satisfies ε-differential privacy, where n is the number of marginal tables. In general $n = C_d^k$, which means all the marginal table cover sets combined by the k-dimensional table are released, in this method $ESE_d = 2^k \times (C_d^k/\varepsilon)$

There is a large amount of redundant information between the marginal tables of the same dimension that is often unnecessary to be published in practice. At the same time, the larger the number of the marginal table, the huger Laplace noise will be mixed. Therefore, in the PriView algorithm, Wahbeh Qardaji et al proposed the marginal table dimension and quantity selection method based on the overlay design to complete the construction of the marginal table cover set. This method further reduce the number of marginal tables n under the cover requirement by reducing the cover relationship between the same dimension tables, which reduces the overall noise. The expected variance of the method is $ESE_d = 2^k \times (n/\varepsilon)$, where $n < C_d^k$.

Issue of Former Method The PriView algorithm proposed by Wahbeh Qardaji et al completed the optimization of the number of marginal tables by the overlay design, but the method still has the following problems.

(1) PriView only gives the approximate range of the optimal marginal table dimension selection. It does not explicitly give the selection method of the marginal table dimension under different dimension data sets as well as the query cover rate the set can provide under this dimension.

(2) The method uses cover design to find the k-dimensional marginal table cover set, and complete the full cover of the query combination of the $k-1$ dimension table. However, the method is under the condition of the assumption that the relationships among the attributes in the data set are completely independent of each other. Actually, there is always a certain correlation between the attributes in the real data set. These correlations determine the validity of the query combination covered by the marginal table. Obviously, PriView's marginal table lookup method has a certain degree of blindness, and will contain redundant invalid query combinations, which will increase the number of published marginal tables and reduce data availability.

To figure out these two problems, we propose a regular marginal table differential privacy publishing algorithm under frequent item sets in section 3. By analyzing the relationship between the dimension change of the marginal table and the cover rate, the algorithm gives the selection method of the table dimension under different query cover requirements. Meanwhile, this method estimates the support of the marginal table combined with the frequent item mining algorithm and establishes the weighted marginal table set cover model with the support degree. By improving the CMC algorithm, a marginal table cover algorithm based on support degree and query is proposed. Finally, it achieves targeting cover of valid query combinations, and reduces the number of marginal tables that further improves data availability.

(3) PriView only uses the same dimension table to publish the marginal table cover set, which can improve the data availability by sacrificing a part of the query scope. However, the cover method has certain limitations in the application scenario where the data privacy protection is not high and the query range cannot be reduced.

In order to overcome this shortcoming, we propose a differential privacy publishing algorithm based on irregular marginal table partitioning, which uses different dimension marginal tables to form a cover set in section IV. We control the amount of noise mixed in the publishing marginal table as much as possible to make a balance between privacy protection and data availability by constraining the multi-level query rules.

3. RD-Privacy Algorithm

In this section, we present our RD-Privacy (Regular marginal table Differential Privacy releasing) algorithm. Firstly, the basic implementation flow of our algorithm is briefly described. Secondly, by analyzing the upper and lower bounds of the cover, we propose the relationship formula between the marginal table dimension and the cover. Finally, the filtering condition of the candidate table of marginal table is analyzed, and the marginal table covering algorithm is presented based on it.

3.1. Algorithm Overview

The main idea of the proposed RD-Privacy method is to find n k-marginal tables, which can effectively cover the query set. To improve the availability of the published data, we want to reduce n so as to add less noise, under the condition that the noise should obey the Laplacian distribution to satisfy ε-differential privacy.

Assuming that there is a data set D with dimension d, then a marginal table with k $(k < d)$ dimension is a view of data set D (only k attributes are shown) and naturally there are $\binom{d}{k}$ kinds of k marginal tables. Suppose U_D is all the query collection of D, we can obviously obtain that $|U_D| = \sum_{i=1}^{d} \binom{d}{i} = 2^d - 1$. We define m_k is the query collection set of all k-marginal tables and $m_k^i (1 \leq i \leq \binom{d}{k})$ is the query collection set of the i-th k-marginal tables. Then $m_k = \{s : s \subset U_D, |s| \leq k\}$, and $\left|m_k^i\right| = 2^k - 1$.

It is easy to see that finding n k-marginal table equals to discovering subsets of U_D. As shown in Eq. 2, the query cover σ is mainly related to the parameter k. We analyze it in Section 3.2, and propose a dimension selection method based on Eq. 2.

$$\sigma = \frac{\left|\bigcup_{i=1}^{n} m_k^i\right|}{|U_D|} < \frac{n(2^k - 1)}{2^d - 1}. \tag{2}$$

To satisfy differential privacy, we need to add $Laplace(n/\varepsilon)$ noise into the marginal tables. The expected squared error of n marginal tables ESE_n is shown in Eq. 3.

$$ESE_n = 2^{k+1} \times (n/\varepsilon)^2. \tag{3}$$

From Eq. 3, we can see that when n is too large, the publishing middle ware which satisfy ε-differential privacy needs to be mixed into the excessive noise, resulting into lower data availability.

To have a clear understanding of notations, we give a summary of notation of in Table 1.

Table 1. Summary of notations

Name	Notion
D	Database
d	dimension of Dataset D
k	dimension of marginal table
U_D	query collection of Dataset D
m_k^i	query collection of the i-th k marginal tables
m_k	query collection of all the k marginal tables
ESE_n	the Expected Squared Error of n marginal tables
M_{opt}	the optimal solution
M_{app}	the approximate solution of M_{opt}
$m(.)$	function of calculating the marginal benefit
$Mben(.)$	represents the rate of $m(.)$

Since ESE_n can be decreased by reducing n, we propose to obtain smaller size k-marginal tables based on frequent attributes. Firstly, we get the frequent item sets and corresponding support of attributes in data set D, and then use their support to weight marginal tables. By this way, we can filter out the ineffective marginal tables with low and little influence on querying. A more detailed description can be found in Section 3.3.

In the following, we give the definition of our weighted k-marginal tables covering problem.

Definition 1. For a data set D, the query collection U_D and the cover σ, our weighted k-marginal tables covering problem is to find n k-marginal tables, so as to satisfy $\left|\bigcup_{i=1}^{n} m_k^i\right| / |U_D| \geq \sigma$, the support of these k-marginal tables $\sum_{i=1}^{n} Sup(m_k^i)$ as large as possible, the expected squared error ESE_n as small as possible.

This kind of problem has been proved to be NP-hard, so it is hard to find the optimal solution M_{opt} in polynomial time. To address this issue, we propose RD-Privacy algorithm, which improves classic CMC algorithm [32]. Firstly, we generate candidate query collection of the k-marginal table from D. Secondly, we weight each query collection m_k^i by their frequency. After that, the corresponding k-marginal tables are weighted by their query collection. Thirdly, A algorithm FMC (Frequent item sets Marginal table Covering algorithm) is presented to obtain a nearly optimal k-marginal tables through their weights. Finally, after the noising and consistency processing, we can get the released marginal table query middle ware.

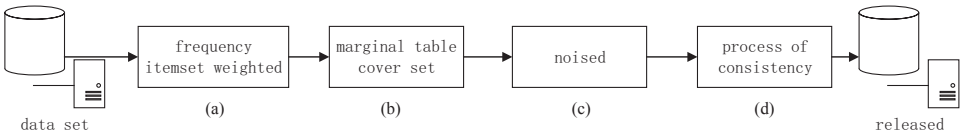

Fig. 1. The flow chart of differential privacy release algorithm based on regular marginal tables

3.2. Selection for the marginal table dimension *k*

The dimension k of marginal table is a main factor to the query cover. In this subsection, we discuss about the selection of k according to σ. The cover of marginal table cover set is σ_k, and the number of count queries that can be performed on n k-dimensional marginal tables. The cover σ_k is shown in Eq. 4.

$$\sigma_k = \frac{\left|\bigcup_{i=1}^{n} \binom{k}{i}\right|}{2^d - 1} \tag{4}$$

Cover Upper Bound It is easy to understand that σ_k can get maximal value when all m_k^i are disjoint as Eq. 5.

$$\sigma_d = \frac{\sum_{i=1}^{k} \binom{d}{i}}{2^d - 1} \tag{5}$$

As it is impossible that all m_k^i are disjoint, then

$$\sum_{i=1}^{k} \binom{d}{i} > (2^d - 1) \times \sigma \tag{6}$$

So, let $f(d,k) = \sum_{i=1}^{k} \binom{d}{i}$, we get Eq. 9 and Eq. 10.

$$f(d,k) = 2^{d-1} \exp^{\frac{(d-2k-2)^2}{4(1+k-d)}}, where\ k \leq \frac{d}{2} \tag{7}$$

$$f(d,k) = 2^{d-1}(2 - \exp^{\frac{(2k-d-2)^2}{4(1-k)}}), where\ \frac{d}{2} < k < d \tag{8}$$

$$\sigma \leq 0.5 * \exp^{\frac{(d-2k-2)^2}{4(1+k-d)}}, where\ k \leq \frac{d}{2} \tag{9}$$

$$\sigma \leq 0.5 * \exp^{2-\frac{(2k-d-2)^2}{4(1-k)}}, where\ \frac{d}{2} < k < d \tag{10}$$

Cover Lower Bound Since the weighted set cover problem is NP-hard, the result m_{app} obtained from FMC algorithm is an approximation of the optimal set m_{opt}, the approximate rate is $1 - 1/e$. The approximate cover is lower than optimal cover. As a result, we need to analyze the lower bound of optimal solution to choose the dimension of marginal table to avoid the loss caused by approximation of cover. The lower bound of $M_{opt}$is a k-marginal table cover set which can cover all k-1 way marginal table query combination, so $\sigma_{opt} > \frac{\sum_{i=1}^{k-1} \binom{d}{i}}{2^d - 1}$. From Eq. 7 and Eq. 8, we can get the lower limit of M_{app} as shown as Eq. 11 and Eq. 12.

$$\sigma_{app} > 0.5(1 - 1/e) * \exp^{\frac{(d-2k)^2}{4(k-d)}}, where\ k \leq \frac{d}{2} \tag{11}$$

$$\sigma_{app} > 0.5 * (1 - 1/e) \exp^{2+\frac{(2k-d-4)^2}{4k}}, where\ \frac{d}{2} < k < d \tag{12}$$

In summary, for a given σ, we can get the marginal table of dimension k according to Eq. 11 and Eq. 12.

We give a verification for the efficiency of k-marginal tables releasing. We use 6 test data whose dimensions are $\{15, 17, 19, 21, 23, 25\}$, and use differential dimensional marginal table to form the cover set to analyze the distribution of cover. The distribution is shown in Fig. 2. We can observe that the cover benefit is getting slower when the dimensions are getting larger. In summary, releasing the marginal table is useful for dimensional reduction.

3.3. Selection of Marginal Table and Error Analysis

After obtaining the dimension of marginal table from σ, we need to filter out the marginal table further. PriView is very useful, but it doesn't take the relationship between attributes

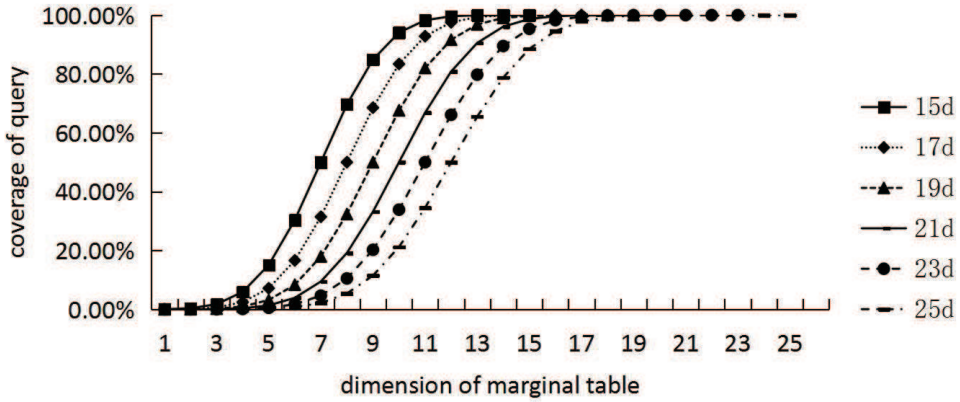

Fig. 2. Cover distribution of different marginal tables under different dimensions

into consideration. To improve this method, we analyze the released data set through frequent item sets. In this method, we can weight the marginal table, and the marginal benefits value of each marginal table are also taken into account. As a result, the marginal table is more reasonable to reduce the redundant marginal table and improve data availability.

Marginal Benefit We use $m(S)$ to represent the marginal benefit of a collection of k-marginal tables S [33]. $Mben(E, S)$ represents the increasing marginal benefit when adding another k-marginal table to S.

$$Mben(E, S) = m(S \cup E) - m(S). \tag{13}$$

It is usually hard to accurately compute $Mben(E, S)$ through Eq. 13. A good news is that $Mben(E, S)$ can be easily computed by the following Eq. 14, if only one k-marginal tables is contained in S.

$$Mben(E, S) = (2^s - 1) + (2^s - 1) \times (2^v - 1) = 2^k - 2^v \tag{14}$$

Where v is the size of the same attributes between E and S. Inspired by Eq. 14, we present a novel approach to approximately compute $Mben(E, S)$ when there are more than one k-marginal tables in S. The approach is shown in Algorithm 4. In order to further verify this approach, we compare the traditional method with the proposed approximate method in Table 4.

FMC. The FMC algorithm contains marginal table generation and a filtering process. In the marginal table preprocessing stage, this paper uses Apriori algorithm to extract the frequent items and attach weight to marginal tables with corresponding support. The algorithm of weighting marginal tables (Weight-MT) is shown in the algorithm 2.

Table 2. Error analysis table of marginal benefit estimation method

Dimension k	6	7	8	9	10
Average error rate	0.167169	0.208876	0.210311	0.207712	0.229283
Accurate Method (ms)	524	899	1691	2524	3107
$Mben_{app}$ (ms)	88	83	98	132	137

Algorithm 2: Weight-MT

Input : original data set D, dimension of marginal table k, min-sup s, N
Output: A_{sup}, M_{sup}

```
Freq = Apriori(D,s)
Mar_k= get k-marginal tables from D
for each mar in Mar_k do
    mar_weight = 0, count = 0
    for each f in Freq do
        if f⊂ any combination of m then
            mar_weight = +f.sup/ |D| ;
        count++;
    mar_weight = mar_weight * (count/n) A_sup.push(mar_weight);
    M_sup.add(mar, m_weight)
```

The dimension of original database D is 17, and we choose k from 6 to 10 to observe the computing time (the first two rows) and the error respectively. From Table 4, it is clear to see that $Mben_{app}$ is computed more faster than traditional method. Meanwhile, the relative error between the estimated value and the actual value is small. Although the relative error increases with the increase of the marginal table dimension, the error rate basically floats around 0.2. In the experiment we find that the trend value of the marginal benefit value obtained by the estimation method is almost the same as the real result, so the effect of the average error rate on the validity of the estimated value can be neglected.

Noise Error Analysis Similar to PriView, we use the expected squared error to evaluate the noise error of RD-privacy, denoted as ESE_{RD} in Eq. 15, where m is the number of the margin tables whose support are smaller than given threshold.

$$ESE_{RD} = 2^{k+1} \times \left(\binom{d}{k} - m/\varepsilon \right)^2 \tag{15}$$

3.4. The Proposed Marginal Table Covering Algorithm

Algorithm Design The weighted k-marginal tables covering problem defined in Definition 1 is similar to a weighted set cover problem. In 2015, Golab proposed a CMC algorithm [30], which is an effective solution to this kind of problem. Inspired by CMC, we proposed the FMC algorithm to solve our problem based on frequent items.

Algorithm 3: FMC

Input : original data set D, dimension of marginal table k,A_{sup},M_{sup},cover σ.
Output: the collection of marginal table Mar_{res}

1 Mar_k= get k-marginal tables from D
2 Mar=$\emptyset$
3 $Mben[mar]$=$\emptyset$
4 **for** each mar in Mar_k **do**
5 $Mben[mar] = Mben(mar, Mar_{res})$
6 **for** i = 1 to $|Mar_k|$ **do**
7 s = get marginal table whose benefit = max($Mben[A_{sup}]$);
8 **if** cover of $Mar_{res} \geq \sigma$ **then**
9 Return;
10 **if** M_{sup}[s] $< \frac{2k-1}{2}$ && $MBEN[s] < sup_t$ **then**
11 Break;
12 delete s from MBen and s in Mar_{res};
13 updateMBen according to algorithm 4

Algorithm 3 is the procedure of FMC. In line 1 to 3, it's the data initialization phase, we get the k marginal table candidate Mar_k from D, and then calculate each marginal benefit of k marginal table to $Mben[mar]$. Line 7 to 13 is the detail process of filtering candidate marginal table. In the beginning , Mar_{res} is empty, then all marginal benefit is the same. So we choose the max support weight marginal table. With the change of Mar_{res}, we update the $Mben[A_{sup}]$ in line 13 according to algorithm 4.

Algorithm 4: MBen

Input : marginal table mar_k, the collection of marginal table Mar_k, the length of marginal table k
Output: the marginal benefits Mben(mar_k,Mar_k)

1 $min = \infty$
2 sum = 0
3 i = 0
4 **for** s in Mar_k **do**
5 v = the number of same attribute in marginal mar_k compared with s
6 $Mben = 2^k - 2^v$
7 **if** $Mben < min$ **then**
8 $min = Mben$
9 $sum+ = 2^v - 1$
10 $i++$
11 return $min - (sum/i)$

After obtain the marginal table cover set Mar_{res}, we can then calculate the size of Mar_{res}. Add noise that follow $Laplace(|Mar_{res}|)$ into marginal tables to make the released middle ware satisfy the ε-differential privacy.

3.5. Consistency Analysis of k-marginal Tables

As the random noise is mixed, the direct publishing method of the marginal table covers the existence of inconsistent query results. Therefore, this section introduces consistency processing algorithms and the corresponding analysis.

We use the same way in PriView to consistency the query middle ware. However, the middle ware obtained by PriView is the query combination cover of the partial dimension contained in the marginal table. The marginal table cover set proposed by the algorithm based on the frequent item sets is completely used for all the query combinations in the marginal table, involving more query combinations. Therefore, for the marginal table cover set handled by the method in the same way, the difference between the privacy properties and the noise error of the published marginal table is different. Therefore, in this section, we analyze the consistency of the process, the marginal of this article to cover the set of differential privacy model of the impact of the query results and noise error.

Differential Privacy Analysis After determining the harmonization process duplicates of each target, each target in accordance with an marginal table is normalized. This process is carried out after the mixing process with Laplace noise, so we need to analyze whether privacy protection has been changed.

In order to determine whether the marginal table middle ware still retains the nature of differential privacy protection, the post-processing inefficiencies of the differential privacy model are introduced here, as shown in Theorem 1. Obviously, since the differential privacy model has post-processing inefficiencies, the subsequent consistency handling operation for the overlay marginal table satisfying the difference privacy does not affect the differential privacy protection feature of the overlay marginal table set itself.

$Theorem1$ (post-processing inefficiencies): If the algorithm S satisfies ε-differential privacy, then for any function, the new processing mechanism $M = \varphi(S(D))$ still satisfies ε-differential privacy.

Noise Error Analysis From previous literature [34] we can get following equation.

$$Q(M_c) = Q(M_c) + \frac{U(A) - Q(M)}{2^{|M|-|A|}} \tag{16}$$

We can know from consistency processing, the first step is to calculate the target value of the same query paradigm A from different marginal tables. Set query results in original data set is $U(A)$ and Laplace noise δ. After noise added process, the result from a k-marginal tables M_i is shown in Eq. 17. If there are j k-marginal tables that satisfy paradigm A, the consistent target is shown in Eq. 17.

$$Q(A) = \mu + \sum_{i=1}^{|M_i|-|A|} \delta_i \left(|A| \leq |M_i|\right) \tag{17}$$

$$U(A) = \mu + \frac{1}{j} \sum_{j=1}^{j} \sum_{i=1}^{|M_i|-|A|} \delta_i \left(|A| \leq |M_i|\right) \tag{18}$$

According to the Eq. 16, Eq. 17 and Eq. 18 we can obtain Eq. 19. Let $\eta = \sum_{i=1}^{|M_i|-|A|} \delta_i$. As the added noise follows Laplace distribution, then $E(\eta) \to 0$. Therefore, from the Eq. 19 we can see that, the whole of noise is reduced and valid of released data is improved while published data also satisfies ε-differential privacy.

$$Q(M_c) = \mu + \sum_{i=1}^{|M_i|-|A|} \delta_i + \frac{\sum_{i=1}^{|M_i|-|A|} \delta_i - \frac{1}{j} \sum_{j=1}^{j} \sum_{i=1}^{|M_i|-|A|} \delta_i}{2^{|M_i|-|A|}} \tag{19}$$

4. IM-Privacy Algorithm

In the previous description, the marginal table differential privacy publishing algorithm under frequent item sets uses many k-dimensional tables of the same dimension to ensure the data availability of the published data set at the expense of a certain cover rate. However, this method has certain limitations in the application scenario where the data privacy protection requirement is not high and the query range has high requirements. To solve this problem, we propose a differential privacy publishing algorithm partitioned by irregular marginal table to improve the PriView model from another perspective in this section. The algorithm composes a cover set by leveraging different marginal tables. By constraining the multi-level query rules, the amount of noise mixed in the publishing marginal table is controlled as much as possible to ensure the balance between privacy protection and data availability without sacrificing the scope of the query.

4.1. Global Sensitivity Analysis

The main idea of the proposed IM-Privacy method (Irregular partition Marginal table differential Privacy release algorithm) is to find n marginal tables that has 1 to d dimensions, which can completely cover all query combinations of D. The method can achieve differential privacy protection through the Laplace mechanism. The premise of the implementation of the Laplace mechanism is that the middle ware F has a clear global sensitivity. The calculation method is as shown in Eq. 20.

$$S(F) = \max_{D_1 D_2} (\sum_{f \in F} |f(D_1) - f(D_2)|) \tag{20}$$

Where D_1 and D_2 are adjacent data sets, and f is arbitrary query on F. If the publishing middle ware M_d composed of multidimensional marginal tables cannot determine the query strategy, the global sensitivity under the middle ware cannot be determined, and the differential privacy protection based on the Laplace mechanism cannot be performed. If the global sensitivity is too high, the noise mixed into the middle ware will be large, which may directly reduce the data availability. To solve this problem, this paper proposes the following constraint on the query strategy of the irregularly divided marginal table cover set. We can use the same-dimensional marginal tables to perform the combined query of the same size. If the query result cannot be obtained with the same-dimensional marginal table, then the result of last-dimension marginal table is used to get the query result. Based on this constraint, the formula for querying the query sensitivity $S(M_d)$ of the middle ware M_d is shown in Eq. 21. Thus, in the worst case, the global sensitivity for the query

middle ware under the constraint is $d+1$, which means the query operation with a $d-1$ attribute combination needs to use the d-dimensional marginal table, and the global sensitivity is D at this time. Therefore, under this constraint, $(S(M_d)/\lambda)$-differential privacy can be satisfied when the Laplace noise distribution is $Laplace(S(M_d)/\lambda)$.

$$S(M_d) = \max(1, C_k^{k-1} + 1) \tag{21}$$

In order to overcome this constraint, we need to further obtain the cover algorithm that satisfies the query policy constraint and minimizes the marginal table cover set ESE to get the irregular marginal table cover set M_d. Therefore, this paper obtains the maximum marginal benefit cover strategy for greedy approximation through the further analysis of ESE, and proposes an approximate optimal marginal table cover algorithm combining this strategy, which can get the approximate solution ρ compared with the optimal cover set M_{d-opt} under the premise of meeting the cover requirement.

4.2. Greedy Approximation Maximum Marginal Benefit Cover Strategy

In order to further improve the availability of the data, we need to find the optimal marginal table cover set that satisfies the query constraint, so that the ESE of the released middle ware is as small as possible. Obviously, the problem is a set cover problem that minimizes the objective function min (ESE). The definition of this minimization problem is shown in Eq. 22, where E is an approximate solution and S is an optimal solution. Since the table of different dimensions is used to form the release middle ware, the overall noise intensity needs to be analyzed. The calculation of ESE_m in this case is as shown in the formula Eq. 23, where $\mid m_i \mid$ represents the number of i-dimensional marginal tables.

$$\min\{ESE(S) : S \in E\} \tag{22}$$

$$ESE_m = 2^d(\sum_{i=1}^{d} \mid m_i \mid)Laplace(S(M_d)/\varepsilon) \tag{23}$$

It is difficult to directly estimate the lower bound of the optimal solution S because variables added are randomly distributed to the marginal table, so we cannot the estimate the approximate ratio of the approximation algorithm. However, it can be seen from Eq. 22 that there are two values affecting the change of ESE_m, the number of covered concentrated marginal tables $\sum_{i=1}^{d} \mid m_i \mid$ and the query sensitivity $S(M_d)$. Therefore, we analyze the marginal benefit value that directly affects the number of marginal tables and query sensitivity, and then transform the ESE minimization problem into the optimization problem of marginal benefit.

In the set cover problem of the marginal table, the k-dimensional marginal table benefit value relative to the cover set M is equal to the number of query combinations covered by the k-dimensional table and not covered by M. For marginal benefit and global sensitivity $S(M_d)$, the description of global sensitivity is a combination of queries covered by a marginal table. Therefore, the marginal benefit is directly proportional to the query sensitivity. In order to guarantee the query scope be completely covered, the cover set

M_d whose global sensitivity is $S(M_d) = d + 1$ needs to be added to the d-dimensional marginal table. Therefor we can further derive Eq. 24 according to the Eq. 23.

$$ESE_m = 2^d(\sum_{i=1}^{d} \mid m_i \mid)Laplace((d+1)/\varepsilon) \tag{24}$$

It can be seen from the ESE estimation formula of the marginal table that, while the published data set dimension d is determined, the overall ESE_m is only related to the number of tables. Thus the smaller the number of covered concentrated marginal tables, the smaller the overall ESE, the higher its data availability. The greater the marginal benefit value, the higher the query cover rate that a single marginal table can provide, the smaller the number of marginal tables when conditions to reach the cover rate are satisfied. As a result, the marginal benefit is inversely proportional to the number of marginal tables.

Therefore, the maximum marginal benefit can be obtained by the approximate solution of the optimal marginal table cover set. The process is as follows. Firstly, we select the marginal table with the largest margin benefit value when filtering the k-dimensional table for the first time. Secondly, the remaining k-dimensional marginal table is iteratively searched until only one table with the marginal benefit value of 1 is left in the dimension. The table with the largest margin benefit value is selected in each iteration. Then enter the $k+1$ dimension search process. This is because the table with the residual margin benefit value of 1 can only complete the cover of itself. Based on the query strategy proposed later of this paper, it can be covered by the $k + 1$ dimension marginal table.

In the choice of the overall idea of the algorithm, since the marginal benefit value of the marginal table changes to satisfy the submodule function, the greedy algorithm is selected to search the marginal table, and the approximate solution of the optimal marginal benefit value is obtained. In summary, this section transforms the ECE minimization problem into the marginal benefit maximization problem, and the optimal cover set S has the marginal benefit value $MBen(S) = 2^d - 1$, which means the optimal set covers all the query combination of d-dimensional data under the constraint query condition. Assuming that the marginal benefit of the approximate solution E obtained by the algorithm is $\sum_{e \in E} MBen(e)$, the approximation rate ρ is as shown in Eq. 25.

$$\rho = \min_{S:min-cov} \frac{\sum_{e \in E} MBen(e)}{MBen(S)} \tag{25}$$

4.3. Approximate Optimal Marginal Table Covering Algorithm

In this section we propose an approximate optimal marginal table covering algorithm to solve the marginal table cover set selection problem of the irregular partitioned differential privacy model. The key of the differential partitioning model with irregular division is to use the marginal table of different dimensions to form the cover set of the query range of the original data set. For a data set whose dimension is d, if we iteratively find the optimal marginal table set on all the 1-to-k-dimensional marginal tables, one iteration needs to be traversed $2^d - 1$ times in the worst case. As the dimension of the data set continues to increase, the number of iterations and traversals per iteration becomes larger, the time complexity we traverse the marginal tables of all the 1 to d dimensions directly

and update the marginal benefit values of the remaining tables will be difficult to estimate. Assuming that the number of iterations is n and the original data set dimension is d, the time complexity of the no-grouped algorithm that is $O(n(2^d-1)^2)$ in the worst case.

In order to reduce the time spent in marginal table traversal, we partition the tables according to their dimensions during algorithm implementation. We find the marginal table with the most marginal benefit in each group, and then iterate the remaining table until the marginal benefit value of the table is 1 or there is no remaining table. The pseudo code of the algorithm is shown in Algorithm 5 whose input is the original data set D, the query cover rate is $\sigma = 1$, and the output is the marginal table cover set that satisfies the cover requirement.

Algorithm 5: IM-Privacy

Input : a data set D, $\mid D \mid = d$, query cover fraction σ
Output: a collection of marginal table M

```
1  M = ∅
2  M_u n = ∅
3  for each i = 1 to d do
4      Margin_i = get i-way marginal tables
5      if i=1 then
6          choose ⌈|D|/2⌉ 1-way marginal tables
7      else
8          for each margin in Margin_i do
9              m = get marginal table with max marginal benefit
10             M.push(m)
11             delete m from Margin_i
12             repeat
13                 update marginal benefit of rest marginal table in Margin_i
14                 get 1 marginal table with max marginal benefit
15             until MBen(m) = 1 or |Margin_i| = 0 ;
16             put uncover marginal tables into M_un
17 for each m in M_un do
18     if MBen(m) ≠ 0 then
19         M.push(m)
20 return M
```

Algorithm 5 is the procedure of IM-Privacy. Line 1 to 5 are the data initialization and preparation phases, where M represents the marginal table cover set and $level$ represents the layer to be divided. We partition the group based on the dimensions of the marginal table. Line 7 is the processing of the 1-dimensional marginal table, and the first $\lceil d/2 \rceil$ marginal tables are directly selected from the 1-dimensional marginal table and added to the cover set M. Lines 11 to 20 are the iterative search process of the marginal table. First, we search the marginal table whose benefit is the highest in the i-th dimension table. Then we iterate the remaining marginal table in the i-th dimension, as shown in lines 13 to 16. Each marginal table is recalculated to figure out the marginal benefit of the remaining

table until the set is empty or only the table remains a marginal benefit value of 1. Lines 17 to 20 check the marginal table with a marginal benefit of 1. If there is a table with a marginal benefit of the relative cover set M greater than 0, it can be directly inserted into M to ensure the final published marginal table. The cover of the query set under the multi-level query strategy is 100%.

4.4. Time Complexity Analysis

The approximate optimal marginal table covering algorithm is the key of the differential privacy publishing algorithm based on the irregular marginal table. We analyze the time complexity of the algorithm. The algorithm consists of two parts, the search process of the 1-to-d-dimensional marginal table and the inspection process of the table with the residual marginal benefit of 1. The marginal table iterative search process is a traversal lookup in the marginal table of groups 1 to d, where the number of tables in the group is $n = C_d^k$. In the lookup process, the time consumption of updating the marginal benefit value of the table is related to the solution set size s of the marginal table cover set, and the time complexity of the process is $O(d \times n \times s)$. For the marginal table set $M_{un}(|M_k| = m)$ with the remaining marginal benefit value of 1, the time complexity of the traversal check is $O(m \times s)$. We can figure out that the overall time complexity of the marginal table algorithm is $O(s \times (d \times n + m))$, where $d \ll n$. So the time complexity is $O(s \times (n + m))$

5. Experiment

In this section, we will compare and analyze the two marginal table-based differential privacy publishing algorithms proposed in this paper. The experiment mainly analyzes the feasibility of differential privacy publishing algorithm based on improved cover set from two aspects: algorithm efficiency and data availability. By comparing with the representative differential privacy model, the advantages of the proposed method in improving data availability are verified. The detail of the hardware used in experiment is shown in table 3. We use public data set MSNBC and Kosarak to design experiments and make performance analysis.

Table 3. Experimental configuration

CPU	i7-6560U CPU @ 2.20GHz 2.21GHz
Memory	DDR4 8GB
Disk	256GB SSD
System image	Windows 10 64-bit operating system

MSNBC [35]: This data set is clicked record of web sites collected from msnbc.com and msn.com which contains 989,818 item sets. Each sequence in the data set corresponds to the page category that a user navigates in 24 hours.

Kosarak [36]: This data set records the click flow information of a Hungarian news web site, with 912,627 items. Each record in the data set represents a combination of news identifier that the user clicks.

Now, we describe the parameters used in the experiments, the privacy default budget $\lambda = 1$ and the dimension of marginal table k is chosen from$\{6, 8\}$ for the default cases.

In the rest of the section, we compare the proposed algorithm with the existing differential privacy algorithm in data availability and efficiency. Firstly, we compare the number of marginal tables generated by RD-privacy with the state-of-the-art method. We analyze the average support of the obtained marginal tables between these two methods, and make the error analysis in the last subsection.

5.1. Results on RD-Privacy

Analysis for Quantity of Marginal Table In this subsection, we select the SNBC to conduct experiment, which has 17-dimensional attributes. We use the Apriori algorithm to mine frequent itemsets (the Apriori algorithm sets the minsup as 0.005).

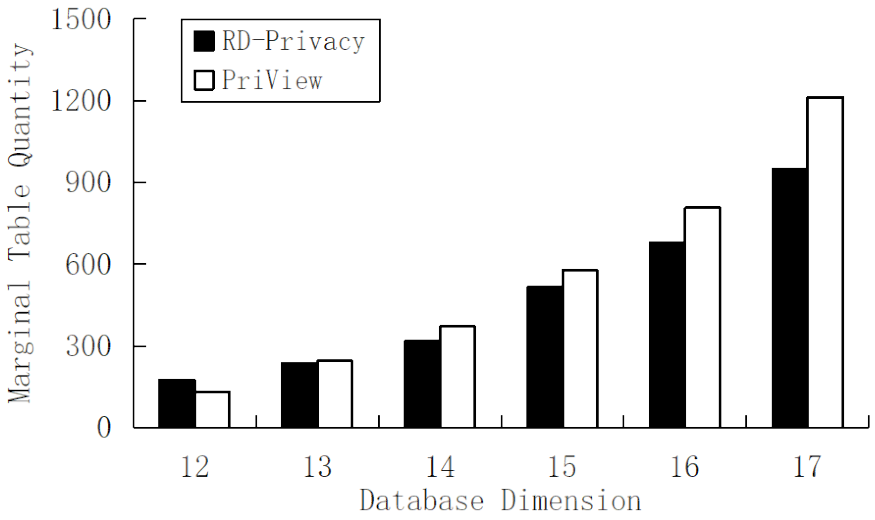

Fig. 3. Marginal table quantity distribution $(k = 6)$

As shown in Fig. 3 and Fig. 4, as the dimension increases, RD-Privacy finds fewer marginal tables than PriView does. The reason is clearly that RD-Privacy make use of Apriori, which can analyze the validation of query combination in MSNBC and delete the combination which have a low support. It is noticeable that when $d = 12$, RD-Privacy obtains more marginal tables because frequent itemsets cannot work better when dimension is low.

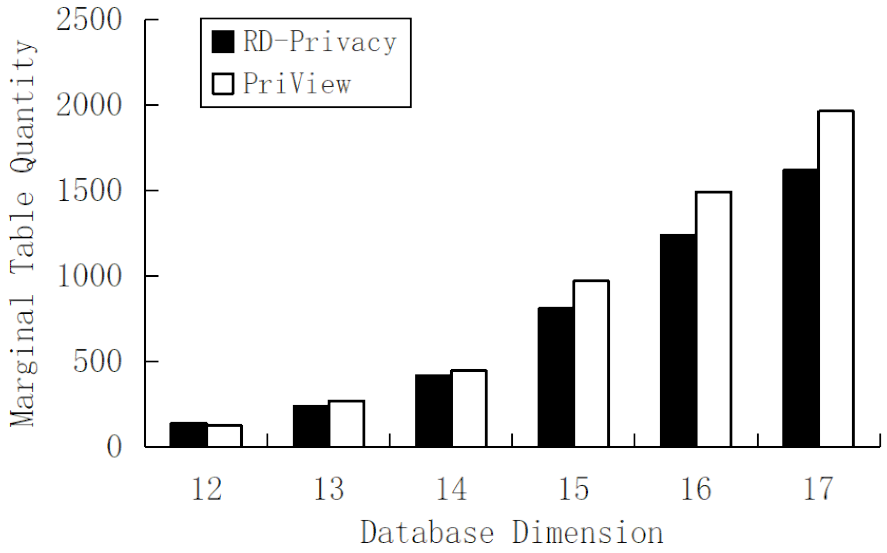

Fig. 4. Marginal table quantity distribution ($k = 8$)

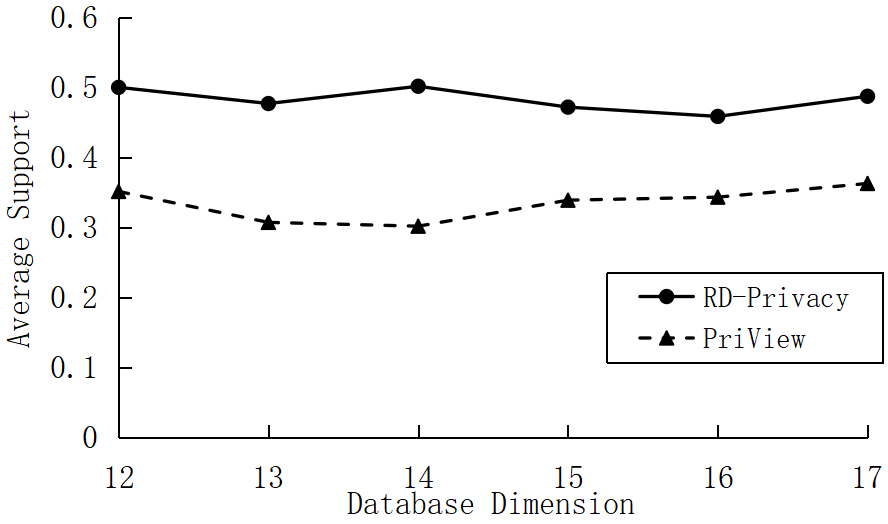

Fig. 5. Average support distribution ($k = 6$)

Analysis of Average Support We can see from Fig. 5 and Fig. 6, the proposed RD-Privacy can get marginal table with lower support which means the obtained marginal tables are more meaningful. When $k = 8$ in Fig. 6, the length of query combination become larger. Then average of support maturely become larger and the Priview still obtain lower support than proposed algorithm.

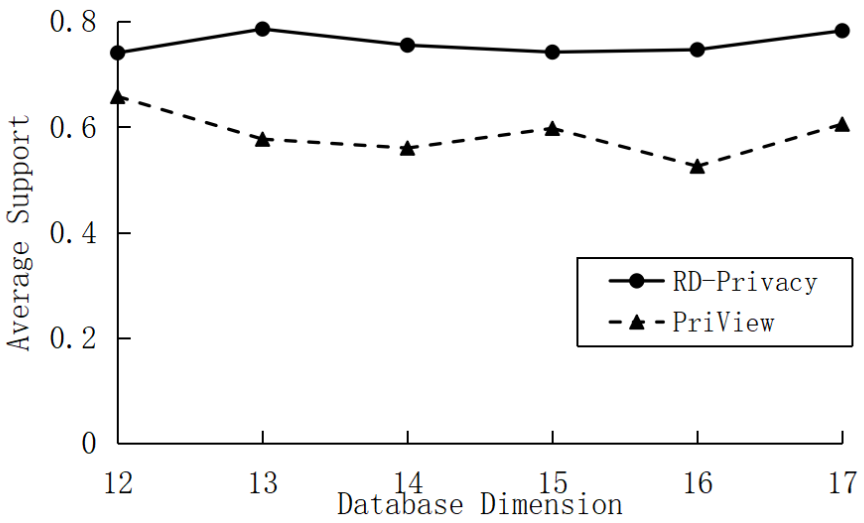

Fig. 6. Average support distribution ($k = 8$)

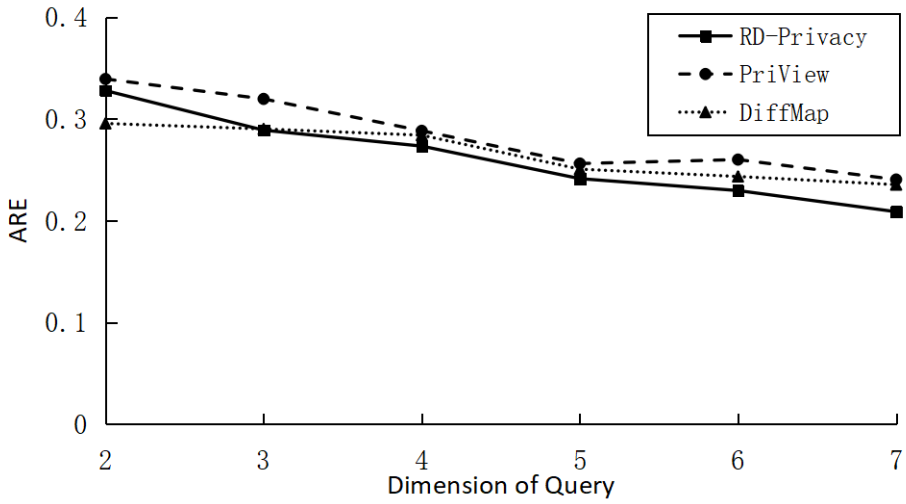

Fig. 7. ARE distribution ($\lambda = 1$)

Analysis of ARE We definite ARE as following Eq. 26, where A is the true value and B is the obtained value.

$$ARE = \frac{|A - B|}{A} \tag{26}$$

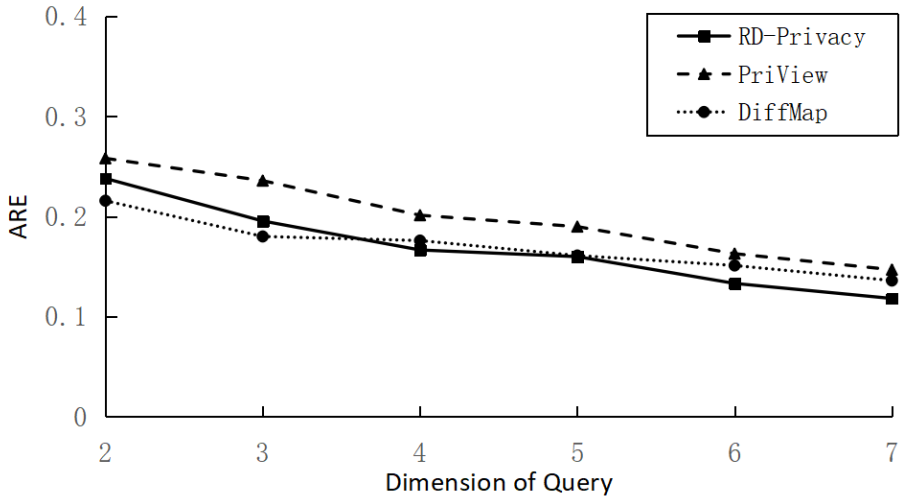

Fig. 8. ARE distribution ($\lambda = 2$)

We can infer from Fig. 7 and Fig. 8 that, in the most of case, the presented algorithm can obtain marginal table with less errors. It is noticeable that when the dimension is small the ARE of RD-Privacy is larger than others. The reason is that the RD-Privacy can obtain more marginal tables based on frequent item sets and then more noises are added.

5.2. Results on IM-Privacy

Experimental Analysis on IM-Privacy This part of the experiment is based on the comparison of IM-Privacy with PINQ and Dwork differential privacy models whose models are differential privacy models with a query cover of 100%. The comparative experiment uses these three algorithms to generate the query middle ware on the Kosarak data set, and then compares the average relative error between the query result and the true value.

The privacy budget in the experiment $\lambda = 1$, the data set used is a user click list of the first 21 categories of news from Kosarak. First, use the IM-Privacy, Dwork, and PINQ for the 21-dimensional Kosarrak data set to generate query middle ware that meets the same differential privacy strength. The IM-Privacy algorithm mainly publishes the middle ware of the marginal table query for the application scenario with high query cover requirements, and the algorithm uses the marginal table of different dimensions to construct the middle ware. There is no limitation of the query dimension by the RD-Privacy algorithm. In the comparative experiment of the algorithm, the setting of the query dimension is different from the former subsection. Due to the marginal table cover set obtained by the IM-Privacy algorithm, the query cover is wide. Therefore, this subsection needs to further investigate the data availability of the query middle ware under the higher query dimension.

In the comparison experiment, in order to verify the data availability under different query dimensions, this paper carries out 5 to 10 random attribute combinations, and each combination distribution performs 100 random count queries. The query paradigm is as

shown in Eq. 27. We record the results of each query with the actual values and then compare the average relative errors of the three different middle ware.

$$SELECT\ COUNT(^*)\ FROM\ D\ WHERE\ A_1 \in S_1\ AND\ A_2 \in S_2 \cdots\ A_m \in S_m \tag{27}$$

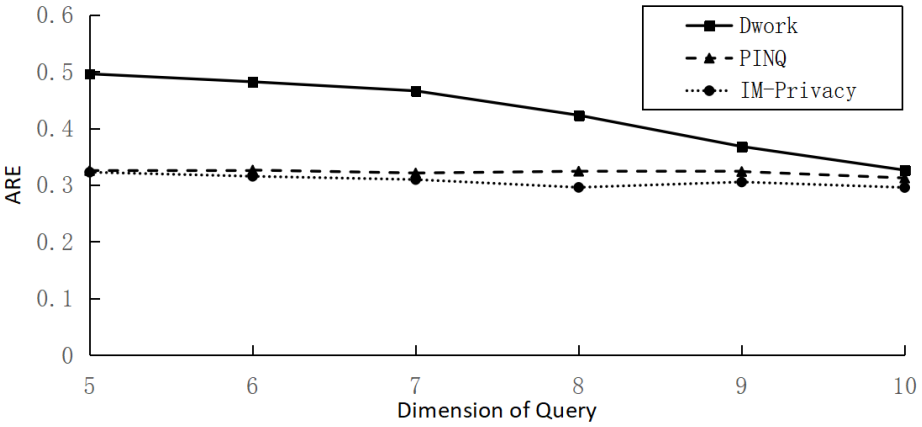

Fig. 9. Comparison chart of ARE for IM-Privacy, PINQ, Dwork

As can be seen from Fig. 9, the query brings more noise when performing a combined count query with a smaller number of attributes since Dwork is a Laplace noise directly added to the contingency table. Therefore, when the query dimensions are 5 to 7, the average relative error of the query results is much higher than the PINQ and the IM-Privacy release algorithm proposed in this paper. PINQ is a way to add noise directly to the query results. IM-Privacy is a rule-based query method (in the k-dimensional marginal table or $(k+1)$-dimensional marginal table with the same size of the combined query). PINQ will only bring the noise in the single query result while IM-Privacy brings noise of two results in the middleware at most, so the average relative error distribution is stable and lower than Dwork's traditional differential privacy model. However, in order to achieve the same level of privacy protection, the global sensitivity of IM-Privacy calculation is lower than half of PINQ, which means the Laplace noise added by IM-Privacy is less than PINQ although the noise of the two results may be brought at most. As a result, the overall average relative error of IM-Privacy is lower than the PINQ. With the increasing of query dimension, the number of items in the Dwork cascade query is gradually reduced. At this time, it is similar to the PINQ method which directly adds noise to the query result. Therefore, when the query dimension is 10, the average relative error of PINQ and Dwork is similar, but the Dwork approach introduces more noise and the query results are more unstable which leads to low data availability. Thus, the proposed method in this paper performs better in improving data availability.

Approximate Rate on IM-Privacy The approximate optimal marginal table covering algorithm in this paper is proposed for the optimal marginal table cover set. Its approximation rate is affected by the size of the data set. Therefore, it is necessary to consider the approximate ratio of the algorithm. We should calculate the ratio between actual irregular marginal table cover set and optimal one in theory. Meanwhile, in order to verify the availability of proposed algorithm, we will also figure out the effect of the change on data set size to approximate ratio stability.

The data set used in the experiment is MSNBC. We select 13 to 17 attributes randomly to form MSNBC of different dimensions and use the IM-Privacy algorithm to find the cover set composed of irregular marginal tables. Finally, we calculate the sum of the marginal benefit values of the marginal table and compare with the theoretical optimal marginal benefit value to analyze the approximation rate.

It can be seen from Table 4 that the overall marginal benefit value obtained by the algorithm is always between 0.8 and 0.9 as the increasing of data dimension, what's more, the ratio is always higher than the theoretical optimal marginal benefit value. It is proved that the approximation algorithm has certain advantages in the stability of its approximation degree and the availabilities of the algorithm.

Table 4. Change of approximation rate for IM-Privacy algorithm

Dimension k	13	14	15	16	17
Optimal marginal benefit	8191	16383	32767	65535	131071
Actual marginal benefit	7127	14376	28881	58169	116467
Approximate rate	0.87010133	0.877494964	0.881405072	0.887602	0.88857947

6. Conclusion and Overlook

The development of the information industry has brought convenience to the office and life of each of us, and it has also created hidden dangers of user data leakage. However, The data publishing process becomes less secure and may result in user privacy leaks. The traditional privacy protection method can not meet the privacy protection requirements in the current environment. Therefore, new differential privacy model is widely used in the data release process with privacy protection requirements. Traditional middle ware for statistical data-based differential privacy algorithms contain more noise and lower data availability due to the differential privacy model that uses the noise to protect data. At present, although the marginal table based differential privacy model effectively reduces the noise, it does not consider the relevance of the attributes in the real data set. Meanwhile, the table cover set contains some invalid query combinations, which reduces the data availability. In the mean time, only the table of the same dimension is selected, which can not meet the needs of practical applications.

To settle these issues, this paper proposes a differential privacy publishing algorithm for regular marginal table differential (RD-Privacy) and irregular marginal table (IM-Privacy) under frequent item sets.

The differential privacy releasing algorithm based on regular marginal tables under frequent item sets uses Apriori algorithm to analyze the actual application data set, comprehensively considers the marginal table support degree and marginal benefit, and provides targeted cover for effective query combination, further improving the data availability of middle ware of the query.

Considering on the low requirements of data privacy protection but high requirements of the cover, a differential privacy publishing algorithm based on irregular marginal table partitioning is proposed. Using the approximate optimal marginal table covering algorithm proposed in this paper, we find that the multi-level edge is satisfied. The table query cover set of the table query policy constraint achieves a balance between privacy protection and data availability to a certain extent.

The paper only studies the differential privacy model under the count query, we can further expand the algorithm to the field of subgraph area. The paper focuses on the research of differential privacy protection for numerical statistical data. We can further study the statistical data of multi-category and multi-valued attributes, and obtain a more applicable method of constructing the marginal table differential privacy model.

Acknowledgment Supported by the Double-class Construction Innovation Project 01431-90518. Supported by the National Nature Science Foundation of China 61370198, 6137-0199, 61672379 and 61300187. Supported by the Liaoning Provincial Natural Science Foundation of China NO. 2019-MS-028.

References

1. Lyu, M., Su, D., Li, N.: Understanding the Sparse Vector Technique for Differential Privacy, PVLDB, 637-648.(2017)
2. Kong, W., Lei, Y., Ma, J.: Data security and privacy information challenges in cloud computing, International Journal of Computational Science and Engineering, 16(3), 212-215.(2018)
3. Shynu, P. G., Singh, K. J.: Privacy preserving secret key extraction protocol for multi-authority attribute-based encryption techniques in cloud computing, International Journal of Embedded Systems, 10(4), 287-300.(2018)
4. Dwork, C.: Differential Privacy, ICALP, 63(6), 1-12.(2006)
5. Atzori, M.: Weak k-anonymity: A low-distortion model for protecting privacy, International Conference on Information Security, 60-71.(2006)
6. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression, (1998)
7. Omran, E., Bokma, A., Abu-Almaati, S.: A k-anonymity based semantic model for protecting personal information and privacy, Advance Computing Conference, 2009. IACC 2009. IEEE International, 1443-1447.(2009)
8. Sweeney, L.: K-anonymity: A model for protecting privacy, Int. J. Uncertain. Fuzziness Knowl.-Based Syst, 10(5), 557-570.(2014)
9. Han, J., Yu, H., Yu, J.: An improved l-diversity model for numerical sensitive attributes, 2008 Third International Conference on Communications and Networking in China, 938-943.(2008)
10. Kroc, L., Sabharwal, A., Selman, B.: Leveraging belief propagation, backtrack search, and statistics for model counting, International Conference on Integration of Ai and Or Techniques in Constraint Programming for Combinatorial Optimization Problems, 127-141.(2008)
11. Ohkubo, J.: Basics of counting statistics, Ieice Transactions on Communications, E96.B(112014), 2733-2740.(2013)

12. Wang, H., Chen, Y., Li, L., Pan, H., Gu, X., Liang, Z.: A novel colon wall flattening model for computed tomographic colonography: Method and validation, Comput Methods Biomech Biomed Eng Imaging Vis, 13(4), 1-14.(2014)
13. Li, C., Hay, M., Rastogi, V., Miklau, G., Mcgregor, A.: Optimizing linear counting queries under differential privacy, Twenty-Ninth ACM Sigmod-Sigact-Sigart Symposium on Principles of Database Systems, PODS 2010, June 6-11, 2010, Indianapolis, Indiana, Usa, 123-134.(2010)
14. Edoh-Alove, E., Bimonte, S., Pinet, F., Bdard, Y.: New design approach to handle spatial vagueness in spatial olap datacubes, International Journal of Agricultural and Environmental Information Systems, 6(3), 29-149.(2015)
15. Dwork, C.: Differential privacy: a survey of results, International Conference on Theory and Applications of MODELS of Computation, 1-19.(2008)
16. Lu, S., Mandava, G., Yan, G., et al.: An exact algorithm for finding cancer driver somatic genome alterations: the weighted mutually exclusive maximum set cover problem, Algorithms for Molecular Biology, 11(1), 1.(2016)
17. Al-Shihabi, S., Arafeh, M., Barghash, M.: An improved hybrid algorithm for the set covering problem, Computers & Industrial Engineering, 85, 328-334.(2015)
18. Cheng, T. C. E., Shafransky, Y., Ng, C. T.: An alternative approach for proving the NP-hardness of optimization problems, European Journal of Operational Research, 248(1), 52-58.(2016)
19. Das, Aparna, Claire M.: A quasipolynomial time approximation scheme for Euclidean capacitated vehicle routing, Algorithmica, 73(1), 115-142.(2015)
20. Feldman, M., Naor, J., Schwartz, R.: A unified continuous greedy algorithm for submodular maximization[C], Foundations of Computer Science (FOCS). IEEE, Palm Springs, CA, USA, 570-579.(2011)
21. Sagnol, G.: Approximation of a maximum-submodular-coverage problem involving spectral functions, with application to experimental designs[J], Discrete Applied Mathematics, 161(1), 258-276.(2013)
22. Wan, P. J., Du, D. Z., Pardalos, P., et al.: Greedy approximations for minimum submodular cover with submodular cost, Computational Optimization and Applications, 45(2), 463-474.(2010)
23. Qardaji, W., Yang, W., Li, N.: Priview:practical differentially private release of marginal contingency tables, 1435-1446.(2014)
24. Zhang, Q., Li, F., Yi, K.: Finding frequent items in probabilistic data, ACM International Conference on Management of Data (SIGMOD), 63(6), 819-832.(2008)
25. Bernecker, T., Kriegel, H. P., Renz, M.,Verhein, F., Zfle, A.: Probabilistic frequent pattern growth for itemset mining in uncertain databases, International Conference on Scientific and Statistical Database Management (SSDBM), 38-55.(2010)
26. Wang, L., Cheung, W. L., Cheng, R., Lee, S. D., Yang, X. S.: Efficient mining of frequent item sets on large uncertain databases, IEEE Transactions on Knowledge and Data Engineering, 2170-2183.(2012)
27. Tong, Y., Chen, L., Cheng, Y., Yu, P. S.: Mining frequent itemsets over uncertain databases, Proceedings of the Very Large Database (VLDB), 1650-1661.(2012)
28. Tang, P., Peterson, E. A.: Mining probabilistic frequent closed itemsets in uncertain databases, Southeast Regional Conference, 86-91.(2011)
29. Agrawal, R., Imieliski, T., Swami, A.: Mining association rules between sets of items in large databases, ACM SIGMOD Record, 22(2), 207-216.(1993)
30. Qardaji, W., Yang, W., Li, N.: PriView: practical differentially private release of marginal contingency tables, Proceedings of the 2014 ACM SIGMOD international conference on Management of data. ACM, Snowbird, UT, USA, 1435-1446.(2014)
31. Fienberg, S. E., Rinaldo, A., Yang, X.: Differential privacy and the risk-utility tradeoff for multi-dimensional contingency tables, International Conference on Privacy in Statistical Databases. Springer Berlin Heidelberg, 187-199.(2010)

32. Golab, L., Korn, F., Li, F., Saha, B., Srivastava, D.: Size-constrained weighted set cover, IEEE, 879-890.(2015)
33. Emek, Y., Rosn, A.: Semi-streaming set cover, International Colloquium on Automata, Languages, and Programming, 453-464.(2014)
34. Hay, M., Rastogi, V., Miklau, G., Dan, S.: Boosting the accuracy of differentially private histograms through consistency, Proceedings of the Vldb Endowment, 3(1-2), 1021-1032.(2009)
35. http://archive.ics.uci.edu/ml/datasets/MSNBC.com+Anonymous+Web+Data
36. Frequent itemset mining data set repository, http://fimi.ua.ac.be/data/.

Haoze Lv is currently in school with the Department of Computer Science in Dalian Maritime University. His research mainly focuses on Big data and privacy protection.

Zhaobin Liu (corresponding author) received the Ph.D degree in computer science from Huazhong University of Science and Technology, China, in 2004. He is currently a Professor in the school of information science and technology, Dalian Maritime University, China. He has been a Senior Visiting Scientist at The University of Auckland, New Zealand, in 2017, and a visiting scholar at University of Central Florida, USA, in 2014 and University of Otago, New Zealand in 2008 respectively. His research interests include big data, cloud computing and data privacy.

Zhonglian Hu is currently pursuing the master's degree with the Department of Computer Science in Dalian Maritime University. His research mainly focuses on Big data and privacy protection.

Lihai Nie is currently an doctoral candidate in the division of Intelligence and Computing, Tianjin University, China. His research interests include data mining, artificial intelligence, network and clouding computing. He has published more than several papers in international journals and conferences.

Weijiang Liu received the Ph.D. degree in computational mathematics from Jilin University in 1998. From June 2004 to June 2006, he was a postdoctoral fellow of Post Doctoral Station for Computer Science and Technology, Southeast University, China. He is currently a professor in School of Information Technology, Dalian Maritime University, China. He has published more than 80 papers and his current research interests include network measurement, network security, and mobile computing.

Xinfeng Ye gained BSc in Computer Science from Hua Qiao University, China, in 1987, and MSc and PhD in Computer Science from The University of Manchester, England, in 1988 and 1991 respectively. He is currently a senior lecturer in the Department of Computer Science at The University of Auckland, New Zealand. His current research interests include cloud computing and system security.

Received: September 15, 2018; Accepted: July 23, 2019.