Short Paper

Publishing Differentially Private Datasets via Stable Microaggregation

Masooma Iftikhar, Qing Wang, Yu Lin
Australian National University, Canberra, Australia
{masooma.iftikhar,qing.wang,yu.lin}@anu.edu.au

ABSTRACT

In recent years, differential privacy has emerged as one formal notion of privacy. Data release based on differential privacy can help researchers to perform statistical analysis on sensitive data of individuals. To publish differentially private datasets, there is a need for preserving data utility, along with data privacy. However, the utility of differentially private datasets is often limited, due to the amount of noise being added to the results of queries. In this paper, we address this issue by proposing a microaggregation-based framework that incorporates microaggregation and differential privacy into the data publishing process. We formulate a new notion of *stable microaggregation* to characterize a desired property of microaggregation and further develop a simple yet effective *stable microaggregation algorithm*. We experimentally verify the utility reduction of our proposed framework on real-world datasets. The experiments show that the proposed framework outperforms the state-of-the-art methods by providing better with-in cluster homogeneity and also reducing noise added into differentially private datasets significantly.

1 INTRODUCTION

Publishing data about individuals often poses a privacy threat because data may contain the sensitive information about individuals, e.g., medical history, and publishing them would intrude upon individual privacy. Thus, to preserve data privacy of individuals, various anonymization techniques have been proposed for data publishing, such as k-anonymity and its extensions [10]. Particularly, with the emerging of differential privacy in recent years [3, 5], a number of works have considered to release differentially private datasets [6, 11]. Such differentially private datasets can guarantee differential privacy controlled by a privacy parameter ε in a robust statistical way.

Broadly speaking, there are two common methods used for generating ε-differentially private datasets in the literature: one is based on differential privacy compliant histograms [11] and the other is based on record perturbation [9]. Histogram-based approaches have some limitations, including: being limited to histogram queries and the exponential growth of the number of histogram bins with the number of attributes [8]. On the other hand, record perturbation based approaches require a large amount of noise being added into the results of queries [9], though these approaches are not limited to histogram queries and allow dealing with any type of attributes.

Nevertheless, when generating differentially private datasets, there always is a trade-off being made between privacy and utility of published data. Ideally, we want to preserve the privacy of individuals while still maintaining the usefulness of data for performing statistical analysis. The utility of ε-differentially private datasets is however limited due to the amount of noise being added to guarantee differential privacy. To enhance the utility of ε-differentially private datasets, in [9] a microaggregation-based mechanism, i.e., *insensitive microaggregation*, has been proposed. It uses microaggregation to achieve k-anonymity in which a certain correspondence between clusters in the microaggregated datasets of two neighboring datasets is imposed. In doing so, the noise added to guarantee differential privacy can be greatly reduced. However, insensitive microaggregation still has the limitations: (1) it yields worse within-cluster homogeneity due to a total order relation required for the distance function [9], and (2) the minimum cluster size k grows with the size n of the dataset and one thus needs $k \geq \sqrt{n}$ to reduce noise.

Contributions. In this paper, we consider the problem of generating ε-differentially private datasets by incorporating microaggregation into the data publishing process. Our work makes the following contributions:

- We present a microaggregation-based framework for generating ε-differentially private datasets and formulate a novel notion of *stable microaggregation* to characterize the correspondence of clusters in microaggregated datasets.
- We propose a stable microaggregation algorithm that can ensure the correspondence of clusters in the microaggregated datasets of two neighboring datasets.
- We experimentally verify the utility reduction of our proposed framework on two real-world datasets containing numerical data. It shows that our algorithm can effectively enhance the utility of released data by providing better within-cluster homogeneity and reducing the amount of noise, in comparison with the state-of-the-art methods.

Related work. Among early works on data anonymization, k-anonymity [10] is a privacy model widely applied to guarantee data privacy of individuals. The popularity of k-anonymity has led to various attempts to address the limitations of k-anonymity [10]. On the other hand, differential privacy [3, 5] is a recent privacy notion that allows statistical analysis of sensitive data while providing strong privacy guarantees. A number of works [8, 9] have combined k-anonymity and differential privacy to enhance the utility of data release. One of these works used microaggregation to achieve k-anonymity, which can reduce the amount of noise added to differentially private datasets [2]. Microaggregation [1] is a family of anonymization algorithms that group similar (homogeneous) records into clusters, then replace each record with its cluster representative. MDAV [2] is the most widely used microaggregation algorithm. The target of a microaggregation algorithm is to yield minimum information loss by maximizing with-in cluster homogeneity. However, the existing works, including MDAV [2] and insensitive microaggregation [9], either produce a low degree of with-in cluster homogeneity or fail to reduce the amount of noise independent of the size of a dataset. Our work in this paper can alleviate both issues.

2 PROBLEM FORMULATION

Let $\mathcal{D}$ be a class of possible datasets. A dataset $X \in \mathcal{D}$ consists of a set of records, each $r_i \in X$ being associated with a set of attributes A. Each individual has only one record in a dataset X.

Definition 2.1. (*Neighboring datasets*) Two datasets $X, Y \in \mathcal{D}$ are said to be *neighboring*, denoted as $X \sim Y$, if $|X| = |Y| = n$, but X and Y differ in one record.

Given a dataset X, we want to generate X_ε (an anonymized version of X) that can provide ε-differential privacy guarantee for protecting the privacy of individuals' records in X.

Definition 2.2. (*Differentially private datasets*) A randomized mechanism $\mathcal{K} : \mathcal{D} \rightarrow \mathcal{D}$ provides ε-differentially private datasets, if for each pair of neighboring datasets $X \sim Y$, and all possible outputs $\mathcal{D}_\varepsilon \subseteq range(\mathcal{K})$, it holds

$$Pr[\mathcal{K}(X) \in \mathcal{D}_\varepsilon] \leq e^\varepsilon \times Pr[\mathcal{K}(Y) \in \mathcal{D}_\varepsilon] \quad (1)$$

where $\varepsilon > 0$ is the differential privacy parameter. Smaller values of ε provide stronger privacy guarantees [4].

ε-differential privacy [3] was originally proposed as a privacy model to protect the responses of interactive queries to a dataset. A query is a function f that extracts data against records in the dataset. A standard way for achieving ε-differential privacy is by adding random noise to the true response of f, and the random noise is calibrated according to the *sensitivity* (Δ) of f, e.g. L_1-sensitivity [5]. For numerical data, the addition of noise can be drawn from a Laplace distribution by first computing the answer $f(X)$ and then generating the noisy answer $f(X) = f(X) + Lap(\Delta(f)/\epsilon)$ to provide ε-differential privacy. Although ε-differential privacy was not initially adapted for the purpose of generating anonymized datasets, but later in [7, 8] differentially private datasets were generated by considering data publishing as the answers to subsequent queries for each record in the dataset.

The L_1-sensitivity of f measures the maximum variation in the query f between two neighboring datasets $X \sim Y$ as follows.

Definition 2.3. (L_1-*sensitivity*) The L_1-sensitivity of a query $f : \mathcal{D} \rightarrow \mathbb{R}^d$ is the smallest number $\Delta(f)$ such that for all neighboring datasets $X \sim Y \in \mathcal{D}$

$$\|f(X) - f(Y)\|_1 \leq \Delta(f), \quad (2)$$

where $\|.\|_1$ denotes the L_1-norm.

Given a dataset X, a microaggregated dataset $\overline{X}$ is created by a microaggregation algorithm $\mathcal{M}$ in two stages. First, X is partitioned into a set of clusters C_X, such that each cluster in C_X has at least k records, where k is a preset constant value, and the records within each cluster are as similar as possible (homogeneous). Second, it aggregates each cluster in C_X by replacing each record with the representative record of the cluster.

In this paper, we aim to generate ε-differentially private datasets by using microaggregation for improving data utility. As illustrated in Figure 1, a microaggregated dataset $\overline{X}$ resulting from running $\mathcal{M}$ over X is added between X and X_ε to increase utility of X_ε. In doing so, the original query f is approximated by $f \circ \mathcal{M}$, since f is run on the microaggregated dataset $\overline{X}$ rather than the original dataset X. This thus introduces two kinds of errors: one is the random noise, which depends on the sensitivity $\Delta(f)$ of query f to guarantee ε-differential privacy, and the other one is due to computing f over $\overline{X}$ instead of X. As will be discussed in Section 4, the first kind of error is much larger than the second kind of error in terms of the information loss in ε-differentially private datasets. To increase the overall utility, the key challenge is how to reduce $\Delta(f \circ \mathcal{M})$ such that $\Delta(f \circ \mathcal{M}) \leq \Delta(f)$.

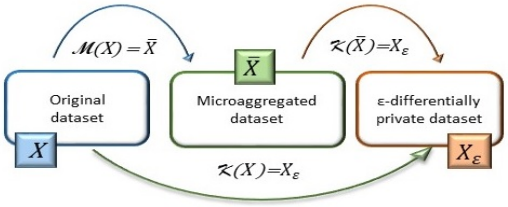

Figure 1: Problem setting.

3 PROPOSED FRAMEWORK

In this section we present the details of the proposed framework.

3.1 Stable Microaggregation

Given $X \sim Y$ that only differ in a single record, their microaggregated datasets $\overline{X}$ and $\overline{Y}$ however may generate considerably different clusters, leading to a much larger $\Delta(f \circ \mathcal{M})$ than $\Delta(f)$. Suppose that we modify a record x in X to x' in Y, i.e., $X \sim Y$. As depicted in Figure 2, a microaggregation algorithm $\mathcal{M}$ (e.g. MDAV [2]) with $k = 4$ can generate C_X and C_Y over X and Y, respectively. Although X and Y only differ in one record, the clusters in C_X and C_Y are completely unrelated. The maximum variation between one cluster from C_X and another unrelated cluster from C_Y is $\Delta(f)$. Since there are n/k clusters in C_X and C_Y, $\Delta(f \circ \mathcal{M}) = n/k \times \Delta(f)$, which can be significantly higher than $\Delta(f)$ when the datasets are large, i.e., n is large.

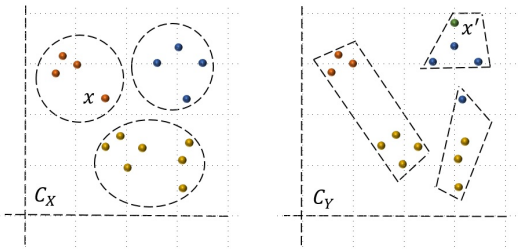

Figure 2: Clusters C_X and C_Y generated by $\mathcal{M}$ over $X \sim Y$.

To address the above issue, the notion of *insensitive* microaggregation was proposed [9]. A microaggregation algorithm $\mathcal{M}$ is said to be *insensitive* if, for every pair of neighboring datasets $X \sim Y$, there is a bijection between C_X and C_Y such that each pair of corresponding clusters differs at most in a single record. This implies that the maximum variation between each pair of corresponding clusters is reduced to $1/k \times \Delta(f)$. Since there are still n/k clusters, $\Delta(f \circ \mathcal{M})$ is $n/k \times \Delta(f)/k$. As a result, insensitive microaggregation may greatly reduce sensitivity as compared with $n/k \times \Delta(f)$ for standard microaggregation.

However, insensitive microaggregation still has some limitations. First, to achieve $\Delta(f \circ \mathcal{M}) \leq \Delta(f)$ as desired, $(n/k \times \Delta(f)/k) \leq \Delta(f)$ must hold. Therefore, one needs $k \geq \sqrt{n}$ in order to reduce added noise in comparison with directly applying $\mathcal{K}$ over X [8]. For large datasets, k thus needs to be large enough for reduced sensitivity. Second, as noted in the work [8] and will also be discussed in Section 4, the clusters generated by insensitive microaggregation are often less homogeneous than the clusters generated by standard microaggregation, such as MDAV [2]. This is because, to ensure the insensitive property,

the distance function used by insensitive microaggregation algorithms must be consistent with the total order relation $\leq_X$ [9]. To alleviate these limitations, we define the notion of *stable* microaggregation.

Definition 3.1. (*Stable microaggregation*) Let $\mathcal{M}$ be a microaggregation algorithm, $C_X = \{c_1, ..., c_n\}$ be the set of clusters that results from running $\mathcal{M}$ on X, and $C_Y = \{c'_1, ..., c'_n\}$ be the set of clusters that results from running $\mathcal{M}$ on Y. $\mathcal{M}$ is *stable* if, for every pair of neighboring datasets $X \sim Y$, there is a bijection between C_X and C_Y such that at most two pairs of corresponding clusters in C_X and C_Y differ in a single record.

Since stable microaggregation affects at most two pairs of corresponding clusters in C_X and C_Y, $\Delta(f \circ \mathcal{M})$ is further reduced to $(2 \times \Delta(f)/k)$ as compared to $(n/k \times \Delta(f)/k)$ for insensitive microaggregation. Thus, when $k \geq 2$, the addition of noise can always be reduced in comparison with directly applying $\mathcal{K}$ over X, regardless of the size of a dataset.

Algorithm 1: *Stable Microaggregation Algorithm*

Input: $X \sim Y$ where $r := X - Y$ and $r' := Y - X$
$\mathcal{M}$: a standard microaggregation algorithm
Output: $\overline{X}, \overline{Y}$

1 $C_X \leftarrow \{c_1, ..., c_n\}$ generated by $\mathcal{M}_p$ over X
2 $C_Y \leftarrow replace(C_X, r, r')$
3 $D, L := \phi$
4 **foreach** $c_i \in C_X$ **do**
5 $\quad D := D \cup \{(dist(r', r_{c_i}), c_i)\}$
6 $d_{min}, c_{min} \leftarrow \mathcal{F}_{min}(D)$
7 **if** $r' \in c_{min}$ **then**
8 $\quad \overline{Y} \leftarrow \mathcal{M}_a(C_Y)$
9 **else**
10 $\quad c_i := c(r')$
11 $\quad D := D - \{(\mathcal{G}_{dist}(D, c_i), c_i)\}$
12 $\quad$ **foreach** $c_j \in C_X \setminus \{c_i\}$ **do**
13 $\quad\quad d_j \leftarrow \mathcal{G}_{dist}(D, c_j)$
14 $\quad\quad D := D - \{(d_j, c_j)\} \cup \{(dist(r_{c_i}, r_{c_j}) + d_j, c_j)\}$
15 $\quad d_{min}, c_{min} \leftarrow \mathcal{F}_{min}(D)$
16 $\quad$ **foreach** $r_i \in c_{min}$ **do**
17 $\quad\quad swap(C_Y, r', r_i)$
18 $\quad\quad \overline{Y}_i \leftarrow \mathcal{M}_a(C_Y)$
19 $\quad\quad L := L \cup \{(\mathcal{I}_{loss}(Y_i, \overline{Y}_i), \overline{Y}_i)\}$
20 $\quad \overline{Y} \leftarrow \mathcal{F}_{min}(L)$
21 $\overline{X} \leftarrow \mathcal{M}_a(C_X)$
22 **Return** $\overline{X}, \overline{Y}$

3.2 Algorithm Description

Our proposed *stable microaggregation algorithm* is described in Algorithm 1. Given $X \sim Y$, we start with partitioning the dataset X into C_X by $\mathcal{M}_p$, i.e., the partition function of a microaggregation algorithm $\mathcal{M}$. Then we replace the record $r \in C_X$ with r' and initialize D and L (Lines 1-3). For each cluster $c_i \in C_X$, by means of function $dist()$, we compute distance between r' and r_{c_i}, where r_{c_i} is the representative record of c_i. Then, we compute d_{min}, i.e., the minimum distance in D, and c_{min}, i.e., the cluster in C_X with d_{min}, by means of $\mathcal{F}_{min}$ function (Lines 4-6). If r' is in the cluster c_{min} of C_Y, then we aggregate C_Y by $\mathcal{M}_a$ that is the aggregation function of the microaggregation algorithm $\mathcal{M}$ (Lines 7-8). In this case, only one pair of corresponding clusters in C_X and C_Y is affected. Otherwise, for each cluster $c_j \in C_X \setminus \{c_i\}$ where $r' \in c_i$, we compute the distance between the representative records of clusters c_i and c_j, i.e., r_{c_i} and r_{c_j}. We proceed with updating D by summing up both distances of the corresponding clusters, excluding the distance of c_i obtained by function $\mathcal{G}_{dist}$ (Lines 10-14). In order to get the cluster c_j that is of the minimum distance from c_i, we find d_{min}, i.e., the minimum distance, and c_{min}, i.e., the cluster $c_j \in C_X$ with d_{min} from D, by means of $\mathcal{F}_{min}$ function. After that, we swap r' in c_i of C_Y with each record r_i in c_{min} of C_Y, and compute C_Y with the minimum information loss by function $\mathcal{I}_{loss}$ (Lines 15-20). In this case, at most two pairs of clusters differ at most in a single record. The algorithm terminates by returning the microaggregated datasets $\overline{X}$ and $\overline{Y}$ that have the minimum information loss.

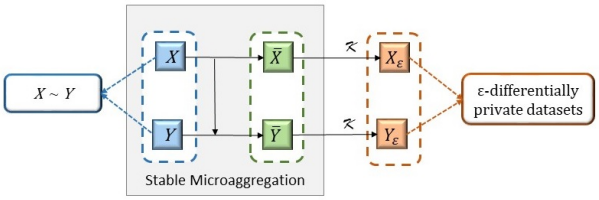

Figure 3: Proposed framework to generate ε-differentially private datasets via stable microaggregation.

A high-level description of our proposed framework is presented in Figure 3, in which stable microaggregation is applied to generate $\overline{X}$ and $\overline{Y}$ by running *Algorithm 1* over $X \sim Y$. Then ε-differentially private datasets X_ε and Y_ε are generated by applying $\mathcal{K}$ over $\overline{X}$ and $\overline{Y}$, respectively.

4 EXPERIMENTS

We evaluated the proposed framework to study how stable microaggregation enhances the utility of differentially private datasets.

Datasets. We used two datasets in the experiments: (1) CENSUS dataset[1] contains 1,080 records [2, 8, 9]. As in [9] we took 4 numerical attributes FEDTAX (Federal income tax liability), FICA (Social security retirement payroll deduction), INTVAL (Amount of interest income) and POTHVAL (Total other persons income). (2) EIA dataset[1] contains 4,092 records [1]. We took 4 numerical attributes attributes RESREVENUE (Revenue from sales to residential consumers), RESSALES (Sales to residential consumers), TOTREVENUE (Revenue from sales to all consumers), and TOTSALES (sales to all consumers).

Following [9], we consider the sensitivity of an attribute to be the difference between the lower bound (i.e. 0) and upper bound (1.5 × the maximum value) of the attribute. For both CENSUS and EIA datasets, the value of k is set to between 2 and 100.

Evaluation measure. We used the measure $IL1s$ [12] to compute the *information loss* between the original and differentially private datasets. Formally, for each record r_i,

$$IL1s = \frac{1}{|A| \cdot n} \sum_{i=1}^{n} \sum_{j=0}^{|A|} \frac{|x_{ij} - x'_{ij}|}{\sqrt{2} S_j} \tag{3}$$

where $|A|$ is the number of attributes, n is the number of records in the dataset, x_{ij} is the value of attribute $a_j \in A$ for record r_i in the original dataset, x'_{ij} is the value of attribute $a_j \in A$ for record

[1] http://neon.vb.cbs.nl/casc/CASCtestsets.htm

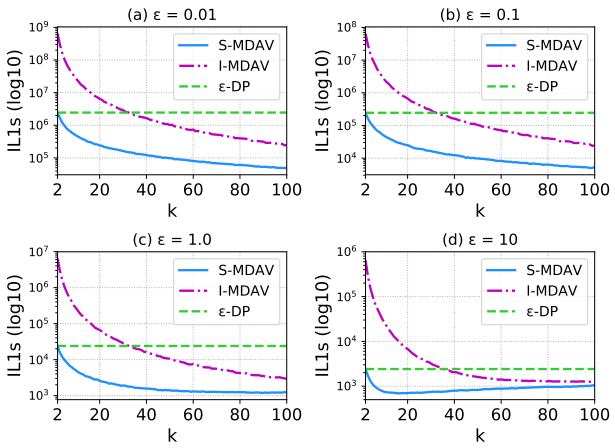

Figure 5: Evolution of $IL1s$ using S-MDAV, I-MDAV and ε-DP for different values of k and ε in CENSUS.

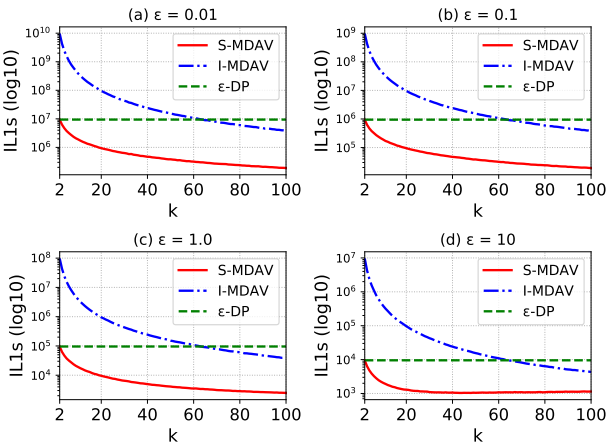

Figure 6: Evolution of $IL1s$ using S-MDAV, I-MDAV and ε-DP for different values of k and ε in EIA.

r_i in the corresponding differentially private dataset, and S_j is the standard deviation of attribute $a_j \in A$ in the original dataset.

Baseline Methods. We considered the following baseline methods: (1) MDAV, which is a standard microaggregation algorithm [2], (2) I-MDAV, which is an insensitive microaggregation algorithm proposed in [9], and (3) ε-DP, which is a standard ε-differential privacy algorithm in which noise is added using the Laplace mechanism [5]. We use S-MDAV to refer to our proposed stable microaggregation algorithm, which extends MDAV in partitioning and aggregation.

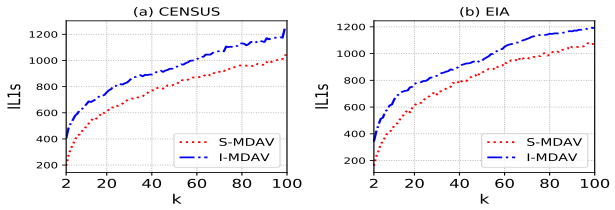

Figure 4: Evolution of $IL1s$ using MDAV and I-MDAV for different values of k: (a) CENSUS and (b) EIA.

Experimental results. We first conducted experiments to compare the information loss of microaggregated datasets that are generated by MDAV and I-MDAV under varying k between 2 to 100. The results are shown in Figure 4. We observe that, for both CENSUS and EIA datasets, the information loss of microaggregated datasets is less with MDAV as compared to I-MDAV. This is because the clusters generated by MDAV are more homogeneous than the clusters generated by I-MDAV. As we used MDAV in our algorithm S-MDAV to generate the clusters in C_X as well as most of the clusters in C_Y, S-MDAV decreases the sensitivity of $f \circ \mathcal{M}$ and thus reduces the errors caused by microaggregation.

Then, to verify the overall utility of ε-differentially private datasets, we conducted experiments to compare the information loss between the original and ε-differentially private datasets generated by using our algorithm S-MDAV and the baseline methods I-MDAV and ε-DP. Figures 5 and 6 present our experimental results. For ε-DP, we used the following privacy parameters $\varepsilon = [0.01, 0.1, 1.0, 10.0]$, which cover the range of differential privacy levels widely used in the literature [4, 7, 8]. For each parameter setting of ε, we ran 3 times and take the average result. The information loss for ε-DP is displayed as horizontal lines, as ε-DP does not depend on k.

Regarding the evolution of $IL1s$ values shown in Figures 5 and 6, we can see that, for every value of ε, I-MDAV is only able to achieve $\Delta(f \circ \mathcal{M}) \leq \Delta(f)$ if $k \geq \sqrt{n}$, i.e., ($k = \sqrt{1,080} \approx 33$ for CENSUS and $k = \sqrt{4,092} \approx 64$ for EIA). This is consistent with the previous discussion in Section 3. Nonetheless, this also means that for large datasets I-MDAV requires k to be enough large in order to effectively reduce $\Delta(f \circ \mathcal{M})$, i.e., the size of k grows with the size of a dataset n. In contrast, for S-MDAV, as stated in Section 3, one needs $k \geq 2$ to reduce $\Delta(f \circ \mathcal{M})$ as compared to ε-DP. As the experiments show that our proposed algorithm S-MDAV leads to less information loss for every value of ε as compared to I-MDAV and ε-DP in both CENSUS and EIA datasets. This is because the sensitivity $\Delta(f \circ \mathcal{M})$ is significantly reduced when S-MDAV is used for microaggregation.

We have also noticed that by approximating a query f to $f \circ \mathcal{M}$ via microaggregation, the errors caused by random noise that depends on the sensitivity of $f \circ \mathcal{M}$ dominate the impact on the utility of differentially private datasets generated via microaggregation, compared to the errors existing between the original and microaggregated datasets.

REFERENCES

[1] Josep Domingo-Ferrer, Antoni Martínez-Ballesté, Josep Maria Mateo-Sanz, and Francesc Sebé. 2006. Efficient multivariate data-oriented microaggregation. *The VLDB Journal* 15, 4 (2006), 355–369.

[2] Josep Domingo-Ferrer and Vicenç Torra. 2005. Ordinal, continuous and heterogeneous k-anonymity through microaggregation. *KDD* 11, 2 (2005), 195–212.

[3] Cynthia Dwork. 2006. Differential Privacy. In *ICALP*. 1–12.

[4] Cynthia Dwork. 2011. A firm foundation for private data analysis. *Communications of the ACM*- 54, 1 (2011), 86–95.

[5] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *TCC*. 265–284.

[6] Ashwin Machanavajjhala, Xi He, and Michael Hay. 2017. Differential privacy in the wild: A tutorial on current practices & open challenges. In *SIGMOD*. 1727–1730.

[7] Ashwin Machanavajjhala, Daniel Kifer, John Abowd, Johannes Gehrke, and Lars Vilhuber. 2008. Privacy: Theory meets practice on the map. In *ICDE*. 277–286.

[8] Jordi Soria-Comas and Josep Domingo-Ferrer. 2018. Differentially private data publishing via optimal univariate microaggregation and record perturbation. *Knowledge-Based Systems* 153 (2018), 78–90.

[9] Jordi Soria-Comas, Josep Domingo-Ferrer, David Sánchez, and Sergio Martínez. 2014. Enhancing data utility in differential privacy via microaggregation-based k-anonymity. *The VLDB Journal* 23, 5 (2014), 771–794.

[10] K Wang, R Chen, BC Fung, and PS Yu. 2010. Privacy-preserving data publishing: A survey on recent developments. *ACM Computing Surveys*- (2010).

[11] Jia Xu, Zhenjie Zhang, Xiaokui Xiao, Yin Yang, and Ge Yu. 2012. Differentially Private Histogram Publication. In *ICDE*. IEEE, 32–43.

[12] William E Yancey, William E Winkler, and Robert H Creecy. 2002. Disclosure risk assessment in perturbative microdata protection. In *Inference control in statistical databases*. 135–152.