Differentially Private Data Sets Based on Microaggregation and Record Perturbation

Jordi Soria-Comas and Josep Domingo-Ferrer

Universitat Rovira i Virgili,
Department of Computer Science and Mathematics
UNESCO Chair in Data Privacy
Av. Països Catalans 26, E-43007 Tarragona, Catalonia
{jordi.soria, josep.domingo}@urv.cat

Abstract. We present an approach to generate differentially private data sets that consists in adding noise to a microaggregated version of the original data set. While this idea has already been proposed in the literature to reduce the data sensitivity and hence the noise required to reach differential privacy, the novelty of our approach is that we focus on the microaggregated data set as the target of protection, rather than focusing on the original data set and viewing the microaggregated data set as a mere intermediate step. As a result, we avoid the complexities inherent to the insensitive microaggregation used in previous contributions and we significantly improve the utility of the data. This claim is supported by theoretical and empirical utility comparisons between our approach and existing approaches.

Keywords: Anonymization, Differential privacy, Microaggregation, Privacy.

1 Introduction

Microdata (that is, information at the individual level) are usually the most convenient type of data for secondary use. However, the risk of disclosure inherent to releasing such detailed information is significant. Traditionally, data were mostly handled by a reduced number of data controllers (e.g. national statistical offices), who had collected them under strong pledges of privacy. In that scenario, reasonable assumptions about the knowledge available to intruders could be made and the methodology for disclosure risk limitation could be adjusted accordingly. Nowadays, the developments in information technology facilitate the collection of personal data. This bounty of data makes it increasingly difficult to make well-grounded assumptions about the side knowledge available to potential intruders [1].

Differential privacy [2] (DP) is a well-known privacy model that gives privacy guarantees without making any assumption on the intruder's side knowledge. In this sense, DP suits well the current scenario with many data controllers. Unlike privacy models designed to protect sets of microdata (e.g. k-anonymity [3], *l*-diversity [4], *t*-closeness [5]), DP was designed to protect the outcomes of interactive queries. However, this limitation was soon overcome with the development of several approaches to release differentially private microdata (DP microdata) [6–10].

The dominant approach to generate DP microdata is based on the computation of DP histograms. However, histogram-based approaches have severe limitations when the number of attributes grows: for fixed attribute granularities, the number of histogram bins grows exponentially with the number of attributes, which has a severe impact on both computational cost and accuracy. To avoid these issues, we propose to generate the DP data set by masking the records in the original data set. Plain independent masking of the records in the original data set is computationally very efficient (its cost is linear on the size of the data set). However, the amount of masking needed to achieve DP is proportional to the sensitivity (the maximum possible variation) of what is being masked, and the sensitivity of an attribute value in a record is large (typically, as large as the attribute domain size). Therefore, a large amount of masking is needed, that results in very substantial information loss.

In this work we describe a record-level perturbation-based approach to generate DP data sets that uses microaggregation to reduce the sensitivity of attribute values and hence the amount of noise required to attain DP. Our approach does not require the use of any specific microaggregation algorithm, but we will choose some microaggregation algorithms for the sake of evaluation. We also compare our results to previous record perturbation approaches. In Section 2 we briefly introduce some basic concepts about DP. In Section 3 we describe our approach to generate DP data sets. In Section 4 we evaluate several microaggregation strategies theoretically and experimentally (by comparing results among them and by comparing results to already existing approaches). Finally, in Section 5 we summarize the conclusions and outline future research avenues.

2 Background on differential privacy

Differential privacy [2] is popular among academics due to the strong privacy guarantees it offers. DP does not rely on assumptions about the side knowledge available to the intruders. Rather, disclosure risk limitation is tackled in a relative manner: the result of any analysis should be similar between data sets that differ in one record. As stated in [11], under DP individuals have no privacy reason to refuse participating in a data set:

> Any given disclosure will be, within a multiplicative factor, just as likely whether or not the individual participates in the database. As a consequence, there is a nominally higher risk to the individual in participating, and only nominal gain to be had by concealing or misrepresenting one's data.

Differential privacy assumes the presence of a trusted party that: (i) holds the data set, (ii) receives the queries submitted by the data users, and (iii) responds to them in a privacy-aware manner. The notion of differential privacy is formalized according to the following definition:

Definition 1 (ϵ-differential privacy). *A randomized function κ gives ϵ-differential privacy (ϵ-DP) if, for all data sets D_1 and D_2 that differ in one record (a.k.a. neighbor data sets), and all $S \subset Range(\kappa)$, we have*

$$\Pr(\kappa(D_1) \in S) \leq \exp(\epsilon) \Pr(\kappa(D_2) \in S).$$

Given a query function f, the goal in differential privacy is to find a randomized function κ_f that satisfies ϵ-DP and approximates f as closely as possible. For the case of numerical queries, κ_f can be obtained via noise addition; that is $\kappa_f(\cdot) = f(\cdot) + N$, where N is a random noise that has been properly adjusted to attain ϵ-DP. The addition of a Laplace distributed noise whose scale has been adjusted to the global sensitivity of the query f is, probably, the most common approach (although other approaches has been proposed [12–14]).

Definition 2 (L_1-sensitivity). *The L_1-sensitivity, Δf, of a function $f : \mathcal{D}^n \to \mathbb{R}^d$ is the maximum variation of f between data sets that differ in one record:*

$$\Delta f = \max_{d(D,D')=1} \|f(D) - f(D')\|_1 .$$

Proposition 1. *Let $f : \mathcal{D}^n \to \mathbb{R}^d$ be a function. The mechanism $\kappa_f(D) = f(D) + (N_1, \ldots, N_d)$, where N_i are drawn i.i.d. from a Laplace$(0, \Delta f/\epsilon)$ distribution, is ϵ-DP.*

3 DP data sets via microaggregation

Let D be the collected data set. Assume that we want to generate D^ϵ –an anonymized version of D– that satisfies ϵ-DP. Let $I_r(D)$ be the query that returns r. We can think of the data set D as the collected answers to the queries $I_r(D)$ for $r \in D$, and we can generate D_ϵ by collecting ϵ-DP responses to $I_r(D)$ for $r \in D$. Such a naive procedure to generate a DP data set is, however, likely to produce a large information loss. In the end, the purpose of DP is to make sure that individual records do not have any significant effect on query responses, which implies that the accuracy of the responses to $I_r(D)$ is necessarily low.

To make perturbative masking viable for the generation of DP data sets, we have to reduce the sensitivity of the queries used. This requires a shift from individual queries to queries that ask for aggregate or statistical information. Along the lines of [15, 9, 16, 10], our proposal is based on microaggregation. In spite of microaggregation being itself a well-known technique in disclosure risk limitation, we use it here with the sole purpose of reducing the sensitivity of the queries. The disclosure risk limitation comes from the enforcement of DP. This change of purpose carries along a change in the traditional way of thinking about microaggregation.

In standard microaggregation, one splits the data set into clusters of at least k records and then replaces the records in each cluster by the cluster centroid, where the minimum value k prevents the cluster from being too representative of any individual in it. In our case, we are also interested in having not too small clusters (in order to limit the impact of individual contributions and hence the sensitivity), but we can relax the requirement of a minimum cluster size. In our case, the total error is the combination of the error introduced by microaggregation and the error due to noise addition; thus, if adding one more record to a cluster produces a large increase in the microaggregation error, it may be preferable to use the smaller cluster. In this work, we think of microaggregation as an algorithm that proceeds in the following two steps:

1. Split the data set into clusters of records.
2. Compute a representative record of each cluster and replace the records in the cluster by it.

To reduce the error introduced by microaggregation, we usually want to generate clusters that are as homogeneous as possible. For the sake of generality, in this section, we do not favor any particular strategy to generate the microaggregation clusters: they can all have the same cardinality or different ones, they can be optimal (maximally homogeneous) or not, randomized or deterministic, etc. However, to be able to analyze the effect of microaggregation on the sensitivity, we need to fix the particular way in which the records in a cluster are combined to generate a record that is representative of the cluster. In this work, we use the mean as aggregation operation (that is, we compute the centroid of the cluster).

The approach we propose is different from those of [15, 9, 16, 10], in that here we consider that *the data set to be protected is the microaggregated one*, rather than the original one. In other words, given an original data set D, we generate $\bar{D}$ by microaggregation of the records in D. From this point on, we discard D and we focus on protecting $\bar{D}$. Hence, the goal is to publish $\bar{D}^{\epsilon}$, a DP version of $\bar{D}$.

The data set $\bar{D}$ acts as a proxy of the original data set D. Thus, when evaluating the utility of $\bar{D}^{\epsilon}$ we need to account for two sources of error: (i) the error due to the microaggregation (that is, the error caused by using $\bar{D}$ as a proxy of D), and (ii) the noise introduced to attain ϵ-DP. The advantage of the proposed approach lies in the fact that the error introduced in the microaggregation step is likely to be more than compensated by the reduction in the noise required to attain DP (compared to the noise that would be required to attain DP directly from the original data set D).

Since the contribution of a record to the centroid is inversely proportional to the cardinality of the corresponding cluster, the centroid sensitivity can be obtained as the record sensitivity divided by the cluster cardinality. This is formalized in the following proposition.

Proposition 2. *Let $C \subset D$ be a cluster of records and let c be the mean of the records in C. Let ΔD be the L_1-sensitivity of a record in D. The L_1-sensitivity of the centroid c is $\Delta c = \Delta D/|C|$.*

Proof. Δc represents the maximum change in c due to an arbitrary change in a single record. Since the maximum change in a single record is ΔD and each record contributes to c, at most, in a proportion of $1/|C|$, the maximum change in c is $\Delta D/|C|$. □

Notice that the sensitivities may differ for centroids of different clusters, because the sensitivity depends on the cluster cardinality. Once the sensitivity of a centroid c is computed, ϵ-DP can be attained by adding a Laplace noise with zero mean and scale $\Delta c/\epsilon$. Since each cluster contains disjoint records, parallel composition applies; thus, by adding Laplace noise independently to each cluster, we obtain the list of ϵ-DP centroids (see Figure 1).

Since each record replaced by the corresponding centroid, each centroid is repeated as many times as there are records in the corresponding cluster. We now explain why in Figure 1 all repetitions of a centroid value are added exactly the same noise. If we added a different random noise to each repetition of the centroid, we would have $|C|$ non-independent DP outcomes each of which has sensitivity $\Delta D/|C|$; hence, by sequential composition, the sensitivity of the list of centroid repetitions in the cluster would be ΔD, which would cancel the benefits of microaggregation. To keep the sensitivity of the centroid repetitions at $\Delta D/|C|$, we must have a single DP centroid value, that is, we must add exactly the same noise to all the repetitions of given centroid. In other words, for each cluster C_i, we take a single draw, n_i, from the $Laplace(0, \frac{\Delta D}{|C_i|\epsilon})$ distribution and use it to mask the $|C_i|$ occurrences of c_i.

$$\begin{tikzcd}
 & \bar{D} & & D_\epsilon & \\
 & c_1 \arrow[rr] & & c_1 + n_1 & n_1 = Laplace(0, \frac{\Delta D}{|C_1|\epsilon}) \\
C_1 & \cdots & \cdots & \cdots & \\
 & c_1 \arrow[rr] & & c_1 + n_1 & \\
 & c_2 \arrow[rr] & & c_2 + n_2 & n_2 = Laplace(0, \frac{\Delta D}{|C_2|\epsilon}) \\
C_2 & \cdots & \cdots & \cdots & \\
 & c_2 \arrow[rr] & & c_2 + n_2 & \\
 & \vdots & \vdots & \vdots & \\
 & c_l \arrow[rr] & & c_l + n_l & n_l = Laplace(0, \frac{\Delta D}{|C_l|\epsilon}) \\
C_l & \cdots & \cdots & \cdots & \\
 & c_l \arrow[rr] & & c_l + n_l &
\end{tikzcd}$$

Fig. 1. Generation of an ϵ-DP data set using record-level microaggregation to reduce the amount of noise required

The procedure to generate an ϵ-DP data set based on record-level microaggregation is formally described in Algorithm 1. The algorithm takes as input parameters the microaggregated data set $\bar{D}$ (whose records consist of the corresponding cluster centroids), the mapping between records in $\bar{D}$ and clusters, and the desired level ϵ of DP. Next, we fix the noise n_i that will be added to all records mapped to each cluster C_i. Finally, we loop through the records in $\bar{D}$ and add to each record the noise that corresponds to the cluster it is mapped to.

Algorithm 1 Procedure to generate an ϵ-DP data set using record-level microaggregation to reduce the amount of noise required

```
Require:
    $\bar{D} = \{r_1, \ldots, r_L\}$: microaggregated data set (each record $r_j$ is the corresponding cluster centroid)
    Mapping $\tau$ between records of $\bar{D}$ and the clusters $C_1, \ldots, C_l$ formed in the microaggregation
    $\epsilon$: desired level of DP
Output
    $\bar{D}^\epsilon$: an $\epsilon$-DP data set

for $i \in \{1, \ldots, l\}$ do
    set $n_i$= random draw from the $Laplace(0, \frac{\Delta D}{|C_i|\epsilon})$ distribution
end for
for  $j \in \{1, \ldots, L\}$ do
    let $C_i := \tau(r_j)$
    set $r_j^\epsilon = r_j + n_i$
end for
return $\bar{D}^\epsilon = \{r_1^\epsilon, \ldots, r_L^\epsilon\}$
```

The procedure depicted in Figure 1 assumes that microaggregation is performed over whole records (either because the data set contains a single attribute or because multivariate microaggregation over all the attributes is used). In the remainder of this section, we generalize the previous procedure to work independently with several individual attributes or subsets of attributes. Essentially, we split the attributes into disjoint subsets, apply the previous procedure independently to each subset, and use sequential composition to determine the overall level of DP.

Let us assume that the microaggregation has been performed independently over the disjoint subsets of attributes $AS_1, \ldots, AS_m$. Sequential composition says that the level of differential privacy from several independent queries accumulates to determine the overall level of DP. As we aim to work independently with each of the subsets AS_i, following sequential composition, we need to split the overall privacy budget, ϵ, among the previous subsets. That is, we fix values $\epsilon_1, \ldots, \epsilon_m$ subject to the restrictions $\epsilon_i \geq 0$ and $\epsilon_1 + \ldots + \epsilon_m = \epsilon$. For each subset AS_i, we apply the procedure in Algorithm 1 to attain ϵ_i-DP. Sequential composition tells that the result is ϵ-DP. This is illustrated in Figure 2 and formalized in Algorithm 2.

4 Evaluation

We evaluate the proposal in Section 3 by fixing several microaggregation strategies and comparing the new proposal to existing methods that are also based on record perturbation [9, 10]. At first sight, the fact that we employ basic microaggregation algorithms rather than (the more restrictive and less utility-preserving)

$$\begin{array}{ccccccc}
 & \bar{D} & & & & \bar{D}_\epsilon & \\
AS_1 & \dots & AS_m & & AS_1 & \dots & AS_m \\
\hline
c^1_{\rho_1(1)} & \cdots & c^m_{\rho_m(1)} & \longrightarrow & c^1_{\rho_1(1)} + n^1_{\rho_1(1)} & \cdots & c^m_{\rho_m(1)} + n^m_{\rho_m(1)} \\
c^1_{\rho_1(2)} & \cdots & c^m_{\rho_m(2)} & \longrightarrow & c^1_{\rho_1(2)} + n^1_{\rho_1(2)} & \cdots & c^m_{\rho_m(2)} + n^m_{\rho_m(2)} \\
\vdots & & \vdots & \vdots & \vdots & & \vdots \\
c^1_{\rho_1(n)} & \dots & c^m_{\rho_m(n)} & \longrightarrow & c^1_{\rho_1(n)} + n^1_{\rho_1(n)} & \dots & c^m_{\rho_m(n)} + n^m_{\rho_m(n)} \\
\hline
\end{array}$$

where $\rho_i(r)$ =cluster number associated to record r

$$n^i_j = Laplace(0, \frac{\Delta AS_i}{(|C^i_j|\epsilon_i})$$

Fig. 2. Generation of an ϵ-DP data set by independently microaggregating the subsets of attributes $AS_1, \dots, AS_m$ and reaching ϵ_i-DP for group AS_i

Algorithm 2 Procedure to generate an ϵ-DP data set by independently microaggregating the groups of attributes $AS_1, \dots, AS_n$ and reaching ϵ_i-DP for group AS_i

```
Require:
    $AS_1, \dots, AS_m$: list of disjoint subsets of attributes
    $\bar{D}$: microaggregated data set, where microaggregation has been independently computed for the projections on each subset of attributes (each record has been replaced by the centroids of the clusters that contain it in each projection)
    $(\tau_1, \dots, \tau_m)$: $\tau_i$ is the mapping between records in $\bar{D}$ and the clusters $C^i_1, \dots, C^i_{l_i}$ computed for the projection $\bar{D}[AS_i]$ of $\bar{D}$ on attribute subset $AS_i$
    $\epsilon_1, \dots, \epsilon_m$: level of DP for attributes $AS_i$ (subject to $\sum \epsilon_i = \epsilon$)
Output
    $\bar{D}^\epsilon$: an $\epsilon$-DP data set

for $i \in \{1, \dots, m\}$ do
    $\bar{D}^\epsilon[AS_i]$ = Algorithm 1($\bar{D}[AS_i], \tau_i, \epsilon_i$)
end for
return $\bar{D}^\epsilon$
```

insensitive microaggregation [9] seems a substantial advantage. Moreover, the method in Section 3 allows adjusting the noise to the size of each cluster.

A difference between the method of Section 3 and the methods in [9, 10] is that the former considers that the data set to be protected is the microaggregated one ($\bar{D}$), whereas the latter aim at protecting the original data set (D). Nonetheless, regardless of the method used, utility must be evaluated in terms of how good is the DP data set D^ϵ as a replacement for the original data set D.

4.1 Evaluated methods

In Section 3 we did not favor any microaggregation strategy. However, the fact is that the microaggregation approach has a significant impact on the utility of the DP data set output by our method. For that reason, empirical results are necessarily tied to a specific microaggregation strategy.

We evaluate the accuracy of our proposal when microaggregation is instantiated with the MDAV algorithm (a heuristic multivariate microaggregation algorithm, [17]) and with individual ranking MDAV microaggregation (which runs independent univariate MDAV microaggregations for each attribute). We have chosen these microaggregation algorithms not only because they are well known, but because they have previously been used to improve the accuracy of DP data sets generated via record perturbation [9, 10].

It is clear, however, that the above-mentioned microaggregation algorithms have some restrictions that limit the accuracy improvements they can offer. An important limitation is that the clusters they generate have a fixed cardinality k (except, maybe, the last cluster, that is of size between k and $2k-1$). However, as noted in Section 3, the method to generate DP data sets described in that section does not require a fixed cluster size, not even a minimum cluster size.

We have evaluated the following DP methods in our comparison:

– MDAV+DP. The method described in Section 3 instantiated with a multivariate MDAV microaggregation of entire records.
– IR_MDAV+DP. The method described in Section 3 instantiated with individual ranking MDAV microaggregation.
– INS+DP (baseline). The method for DP based on insensitive multivariate microaggregation that is described in [9]. This method is a suitable comparison baseline for MDAV+DP because both methods use multivariate microaggregation of entire records.

The method described in [10] could also be considered as a comparison baseline (it would be a good baseline for IR_MDAV+DP, because both are based on individual ranking MDAV microaggregation). However, we skip it because the computation of the sensitivity in [10] is flawed, which leads to overly reducing the noise required to attain DP.

Even if they do not yield DP, the standalone MDAV and IR_MDAV microaggregation algorithms (without subsequent noise addition to attain DP) have also been evaluated. The reason is that using standalone MDAV and IR_MDAV provides an upper bound of the accuracy reachable with MDAV+DP and IR_MDAV+DP, respectively.

4.2 Theoretical evaluation

Although an empirical evaluation is provided further below, we think that a theoretical comparison of some methods, specifically MDAV+DP and IR_MDAV+DP, can yield some important insights.

The following proposition shows that both MDAV+DP and IR_MDAV+DP can yield an ϵ-DP data set by adding the same amount of noise to each attribute.

Proposition 3. *Given a cluster size k used in microaggregation and a target DP level ϵ, both MDAV+DP and IR_MDAV+DP can yield an ϵ-DP data set by adding the same amount of noise to each original attribute.*

Proof. According to Algorithm 1, to attain ϵ-DP with MDAV+DP, we need to add a noise that is distributed according to a $Laplace(0, \varDelta D/\epsilon)$ to each attribute. Assume now we use IR_MDAV+DP instead, and attribute i having sensitivity $\varDelta D_i$ is added noise drawn from $Laplace(0, \varDelta D_i/\epsilon_i)$. Both Laplace distributions are equal when $\varDelta D/\epsilon = \varDelta D_i/\epsilon_i$, which may be enforced by taking

$$\epsilon_i = \epsilon \frac{\varDelta D_i}{\varDelta D}. \tag{1}$$

Since $\varDelta D = \sum \varDelta D_i$, the sum of the ϵ_i amounts to ϵ (as required by the IR_MDAV+DP method). □

The conclusion from the previous proposition is that IR_MDAV+DP (with appropriate ϵ_i) should always be preferred to MDAV+DP: the error due to microaggregation is smaller with IR_MDAV+DP (because less attributes are clustered together) and the error due to noise addition can be made equal. In spite of this result, for the sake of completeness, we will perform the empirical evaluation over both IR_MDAV+DP and MDAV+DP. Actually, we consider two variants of IR_MDAV+DP: IR_MDAV+DP_1 uses the same level of DP for all attributes ($\epsilon_1 = \ldots = \epsilon_m = \epsilon/m$), and IR_MDAV+DP_2 uses the values of ϵ_i given by Expression (1), for $i = 1, \ldots, m$, so that Proposition 3 holds.

4.3 Evaluation data

The empirical evaluation has been performed on the Census data set, which was first used in the "CASC" European project [18] as a reference data set to test and compare statistical disclosure control methods, and was also used in [9]. This data set contains 13 numerical attributes and 1080 records. For the sake of comparability with [9], we focus on 4 attributes: FICA (Social security retirement payroll deduction), FEDTAX (Federal income tax liability), INTVAL (Amount of interest income) and POTHVAL (Total other persons income).

The selected attributes take values above 0 but they are not naturally upper-bounded. Since the L_1-sensitivity is proportional to the sizes of the domains of attributes, we need to upper-bound the domain of each attribute. For the sake of comparability, we use the upper bounds that were used in [9]; that is, we upper-bound the domain of an attribute by 1.5 times the maximum value of the attribute in the data set. The domain bounds on the attributes are also enforced when adding noise to attain DP: the DP masked values are truncated to lie within the fixed bounds.

4.4 Evaluation measures

The evaluation is based on two measures of error: the sum of squared errors (SSE) and the sum of absolute errors (SAE). The SSE is a measure of overall information loss that is commonly used in the evaluation of SDC methods (and particularly in microaggregation). It is computed as

$$SSE = \sum_{i=1,\ldots,n} \sum_{j=1,\ldots,m} (r_{ij} - r^{\epsilon}_{ij})^2$$

where r_{ij} is the value of attribute j in original record r_i and r^{ϵ}_{ij} is value of attribute j in the record r^{ϵ}_i of the DP data set $\bar{D}^{\epsilon}$ that corresponds to r_i.

SAE is similar to SSE but, rather than being based on squared errors, it is based on absolute errors. It is computed as

$$SAE = \sum_{i=1,\ldots,n} \sum_{j=1,\ldots,m} |r_{ij} - r^{\epsilon}_{ij}|.$$

Both measures give an overall estimation of the error in the generated data set but they differ in the relative importance they attach to the magnitude of each difference. In SSE a large error in a single record may have a large overall impact, while in SAE a large error in a single record can be more easily compensated by small errors in other records.

4.5 Experimental results

Figure 3 shows the evolution of SSE as a function of the cluster size. In both graphs of the figure we can see that, as expected, the SSE for the microaggregation algorithms MDAV and IR_MDAV increases with the size of the cluster (which is represented in the abscissae). There is a steep increase for small cluster sizes that flattens out progressively as the cluster size gets larger. On the contrary, for MDAV+DP and IR_MDAV+DP the opposite occurs: SSE decreases with the size of the clusters and the decrease is steeper for small cluster sizes. We observe that, for large cluster sizes, the SSE of all DP methods converge to the SSE of the underlying microaggregation. This result was to be expected because, the greater the cluster size, the less noise we need to attain DP. As it can be seen by comparing both graphics, the rate of convergence is proportional to ϵ (faster convergence for larger ϵ). The comparison between MDAV_DP and IR_MDAV+DP (both variants) shows that IR_MDAV+DP has a lower SSE. This could also be expected, because IR_MDAV is more utility-preserving than MDAV. The comparison between IR_MDAV+DP_1 and IR_MDAV+DP_2 shows that IR_MDAV+DP_2 has slightly less SSE than IR_MDAV+DP_1, but the difference seems to be relatively small.

We then compared the SSE obtained with the methods in this paper with the SSE obtained with the method in [9]. Figure 4a in [9] shows the SSE of the DP data set generated by performing a prior insensitive microaggregation to reduce the noise needed to reach DP. By comparing that figure with Figure 3, we observe that IR_MDAV+DP with $\epsilon = 1$ performs as well as the insensitive approach in [9] with $\epsilon = 10$. This is a very significant improvement in the utility of the data.

Figure 4 shows the SAE of MDAV+DP, IR_MDAV+DP_1 and IR_MDAV+DP_2, and compares them with the baseline MDAV and IR microaggregation algorithms. Consistently with the theoretical comparison between MDAV+DP and

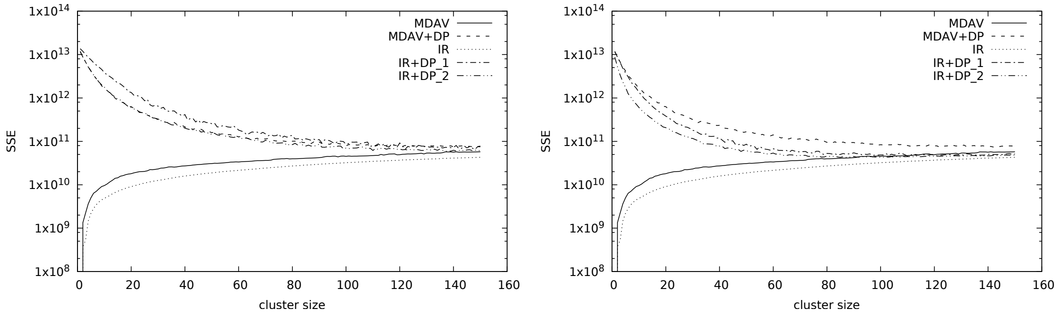

Fig. 3. SSE for $\epsilon = 1$ (left) and $\epsilon = 2$ (right)

IR_MDAV+DP above and with the SSE results, we observe that IR_MDAV+DP is more utility-preserving.

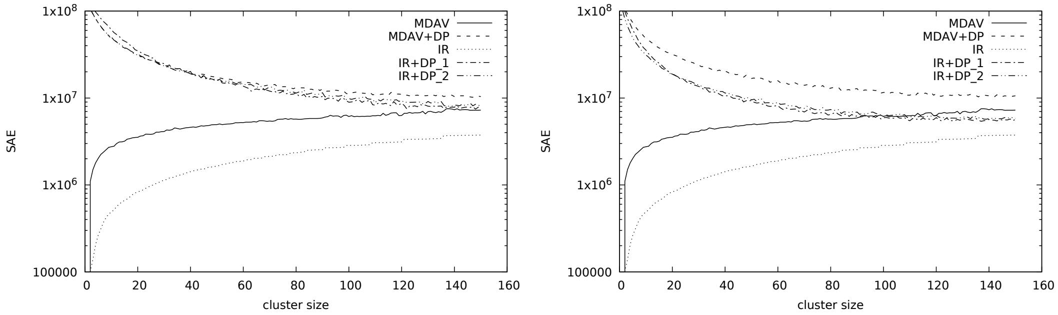

Fig. 4. SAE for $\epsilon = 1$ (left) and $\epsilon = 2$ (right)

5 Conclusions and future work

We have presented an approach to generate DP data sets that consists of adding noise to a microaggregated version of the original data set. Using microaggregation as a prior step to reduce the sensitivity of the data and hence the noise that needs to be added to reach DP had already been proposed in the literature. However, the novelty of our approach is that we focus on the microaggregated data set as the target of protection, rather than focusing on the original data set and viewing the microaggregated data set as a mere intermediate step. As a result, we avoid the complexities inherent to insensitive microaggregation and significantly improve the utility of the data.

The approach we have presented works with any microaggregation algorithm. For concreteness and convenience, we have analyzed three actual approaches to generate DP data sets: MDAV_DP and two variants of IR_MDAV_DP. The comparison (both theoretical and empirical) has shown that IR_MDAV_DP is better than MDAV_DP. Comparisons of IR_MDAV_DP with the insensitive based approach in [9] have shown that IR_MDAV_DP with $\epsilon = 1$ is similar in terms of SSE to the insensitive approach with $\epsilon = 10$. This is a significant improvement in the utility with respect to prior work.

Future work will include:

- Considering non-numerical data by using microaggregation algorithms capable of dealing with categorical data (ordinal, nominal or hierarchical).
- Trying aggregation operators different from the mean (e.g. the medoid) to compute the representative record of a cluster.
- Using variable-size microaggregation heuristics, such as [19], without minimum cluster size (that is, taking $k = 1$). The optimal solution to standalone variable-size microaggregation without minimum cluster size consists of all clusters containing a single record. However, the optimal solution when variable-size microaggregation is used as a preliminary step of DP is likely to contain larger clusters (because larger clusters reduce the noise that is needed to attain DP). In general, the less restrictive nature of variable-size microaggregation algorithms can be expected to deliver DP data sets with better utility, at the cost of increasing the computational effort.

Acknowledgments and disclaimer

Partial support to this work has been received from the European Commission (projects H2020-644024 “CLARUS” and H2020-700540 “CANVAS”), the Government of Catalonia (ICREA Acadèmia Prize to J. Domingo-Ferrer), and from the Spanish Government (projects TIN2014-57364-C2-1-R “SmartGlacis” and TIN 2015-70054-REDC). The authors are with the UNESCO Chair in Data Privacy, but the views in this paper are their own and are not necessarily shared by UNESCO.

References

1. J. Soria-Comas and J. Domingo-Ferrer. Big data privacy: challenges to privacy principles and models. *Data Science and Engineering*, 1(1):21–28, 2015.
2. C. Dwork, F. McSherry, K. Nissim, and A. D. Smith. Calibrating noise to sensitivity in private data analysis. In *Third Theory of Cryptography Conference-TCC 2006*, LNCS 3876, pp. 265–284. Springer, 2006.
3. P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report, SRI International, 1998.
4. A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. l-diversity: privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data*, 1(1), March 2007.

5. N. Li, T. Li, and S. Venkatasubramanian. t-closeness: privacy beyond k-anonymity and l-diversity. In *23th IEEE International Conference on Data Engineering-ICDE 2007*, pp. 106–115. IEEE, 2007.
6. A. Machanavajjhala, D. Kifer, J. Abowd, J. Gehrke, and L. Vilhuber. Privacy: theory meets practice on the map. In *24th IEEE International Conference on Data Engineering-ICDE 2008*, pp. 277–286, 2008.
7. J. Zhang, G. Cormode, C. M. Procopiuc, D. Srivastava, and X. Xiao. Privbayes: private data release via bayesian networks. In *2014 ACM SIGMOD International Conference on Management of Data-SIGMOD '14*, pp. 1423–1434, New York, NY, USA, 2014. ACM.
8. Y. Xiao, L. Xiong, and C. Yuan. Differentially private data release through multidimensional partitioning. In *Secure Data Management*, LNCS 6358, pp. 150–168. Springer, 2010.
9. J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and S. Martínez. Enhancing data utility in differential privacy via microaggregation-based k-anonymity. *The VLDB Journal*, 23(5):771–794, 2014.
10. D. Sánchez, J. Domingo-Ferrer, S. Martínez, and J. Soria-Comas. Utility-preserving differentially private data releases via individual ranking microaggregation. *Information Fusion*, 30:1 – 14, 2016.
11. C. Dwork. Differential privacy. In *Automata, Languages and Programming-ICALP 2006*, LNCS 4052, pp. 1–12. Springer, 2006.
12. J. Soria-Comas and J. Domingo-Ferrer. Optimal data-independent noise for differential privacy. *Information Sciences*, 250:200–214, 2013.
13. F. McSherry and K. Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science-FOCS 2007*, pp. 94–103, Washington DC, 2007. IEEE Computer Society.
14. K. Nissim, S. Raskhodnikova, and A. Smith. Smooth sensitivity and sampling in private data analysis. In *39th Annual ACM Symposium on Theory of Computing-STOC 2007*, pp. 75–84, New York, NY, USA, 2007. ACM.
15. J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and S. Martínez. Improving the utility of differentially private data releases via k-anonymity. In *12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications-TrustCom 2013*, pp. 372–379, 2013.
16. D. Sánchez, J. Domingo-Ferrer, and S. Martínez. Improving the utility of differential privacy via univariate microaggregation. In *Privacy in Statistical Databases-PSD 2014*, LNCS 8744, pp. 130–142. Springer, 2014.
17. J. Domingo-Ferrer and V. Torra. Ordinal, continuous and heterogeneous k-anonymity through microaggregation. *Data Mining and Knowledge Discovery*, 11(2):195-212, 2005.
18. R. Brand, J. Domingo-Ferrer and J. M. Mateo-Sanz. Reference data sets to test and compare SDC methods for the protection of numerical microdata. Deliverable of the EU FP5 "CASC" project, 2002. `http://neon.vb.cbs.nl/casc/CASCtestsets.htm`
19. J. Domingo-Ferrer, F. Sebé and A. Solanas. A polynomial-time approximation to optimal multivariate microaggregation. *Computers & Mathematics with Applications*, 55(4):714-732, 2008.