H. Langweg, M. Meier, B.C. Witt, D. Reinhardt (Hrsg.): Sicherheit 2018,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2018 69

Improving Anonymization Clustering

Florian Thaeter[1] Rüdiger Reischuk[2]

Abstract:

Microaggregation is a technique to preserve privacy when confidential information about individuals shall be used by third parties. A basic property to be established is called k-anonymity. It requires that identifying information about individuals should not be unique, instead there has to be a group of size at least k that looks identical. This is achieved by clustering individuals into appropriate groups and then averaging the identifying information. The question arises how to select these groups such that the information loss by averaging is minimal. This problem has been shown to be NP-hard. Thus, several heuristics called `MDAV` ,`V-MDAV` , ... have been proposed for finding at least a suboptimal clustering.

This paper proposes a more sophisticated, but still efficient strategy called `MDAV`* to construct a good clustering. The question whether to extend a group locally by individuals close by or to start a new group with such individuals is investigated in more depth. This way, a noticeable lower information loss can be achieved which is shown by applying `MDAV`* to several established benchmarks of real data and also to specifically designed random data.

Keywords: Microdata anonymization; k-Anonymity; Microaggregation; group clustering

1 Introduction

We consider databases $\mathcal{X}$ containing n individuals that are characterized by quasi-identifiers and confidential attributes. Quasi-identifiers deliver information that can identify a person, for example date and location of birth. Confidential attributes contain sensitive information about a person, for example the amount of his income or his current diseases that should not be disclosed, more precisely should not be linked to an individual.

In order to discuss algorithmic solutions to the anonymization problem, a precise mathematical model for this setting is needed that will be given first. For a sequence of m quasi-identifiers the set of possible values is given by a cartesian product $QI := QI_1 \times \cdots \times QI_m$, where QI_i denotes the values the i-th quasi-identifier can take. Similarly, for a sequence of p confidential attributes we define $CA := CA_1 \times \cdots \times CA_p$. The database $\mathcal{X}$ consists of n records

[1] Universität zu Lübeck, Institut für Theoretische Informatik, Ratzeburger Allee 160, 23562 Lübeck, Deutschland
thaeter@tcs.uni-luebeck.de
[2] Universität zu Lübeck, Institut für Theoretische Informatik, Ratzeburger Allee 160, 23562 Lübeck, Deutschland
reischuk@tcs.uni-luebeck.de

doi:10.18420/sicherheit2018_05

$(\mathbf{x}_i, \mathbf{y}_i) \in QI \times CA$ with $i \in [1 \dots n]$, where n denotes the number of individuals. The vector $\mathbf{x}_i := (x_i^1, x_i^2, \dots, x_i^m) \in QI$ denotes the quasi-identifiers and $\mathbf{y}_i := (y_i^1, y_i^2, \dots, y_i^p) \in CA$ the confidential attributes of the i-th individual. Restricting $\mathcal{X}$ to the quasi-identifiers we write $\mathcal{X}_{QI} := (\mathbf{x}_1, \dots, \mathbf{x}_n)$ and analogously $\mathcal{X}_{CA} := (\mathbf{y}_1, \dots, \mathbf{y}_n)$ for the confidential attributes. The removal of the i-th individual from $\mathcal{X}$ will be denoted by $\mathcal{X} - (\mathbf{x}_i, \mathbf{y}_i)$.

Such non-public databases may be used by third parties to investigate relations between quasi-identifiers and confidential attributes, for example how the age of a person has influence on his diseases. But the database owner should not simply provide all the tuples $(\mathbf{x}_i, \mathbf{y}_i)$ with their real values because this could violate the privacy of the individuals. Instead, the data has to be anonymized first which is done by an anonymization algorithm μ.

Definition 1.1. Let $\mathcal{X}_\mathbf{x} := \{i \mid \mathbf{x}_i = \mathbf{x}\}$ be the index set of all individuals in $\mathcal{X}$ whose QI value is $\mathbf{x}$. A database $\mathcal{X}$ is k-anonymous if for all $\mathbf{x} \in QI \quad |\mathcal{X}_\mathbf{x}| \geq k$ or $\mathcal{X}_\mathbf{x} = \emptyset$ [S02].

This condition means that every vector $\mathbf{x} \in QI$ contained in $\mathcal{X}$ has to occur with multiplicity at least k. In the special case $k = n$ all values for the quasi-identifiers have to be identical and thus provide no additional information if the set of individuals in $\mathcal{X}$ is known. Such a database guaranteeing maximum privacy will be called *QI-uniform*.

Definition 1.2. An anonymization algorithm μ is a mapping $\mu : (QI \times CA)^n \to (QI \times CA)^n$,

$$\mu : \mathcal{X} := (\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_n, \mathbf{y}_n) \mapsto \hat{\mathcal{X}} := (\hat{\mathbf{x}}_1, \mathbf{y}_1), \dots, (\hat{\mathbf{x}}_n, \mathbf{y}_n)\,,$$

that changes the values of the quasi-identifiers in a database $\mathcal{X}$ to generate an anonymous database $\hat{\mathcal{X}}$. μ achieves k-anonymity if $\mu(\mathcal{X})$ is k-anonymous for every $\mathcal{X}$. If $\mathcal{X}$ is already QI-uniform we require that μ does not change $\mathcal{X}$ to exclude trivial mappings.

The computation of the new values $\hat{\mathbf{x}}_i$ is often done with the help of a *centering algorithm* c. In case of real-valued or ordered data c might be the arithmetic mean denoted by $c_{\texttt{MEAN}}$ or the median $c_{\texttt{MEDIAN}}$ taken of every coordinate independently. $c_{\texttt{MEAN}}(\mathcal{X})$ is also called the *centroid* of $\mathcal{X}$.

Definition 1.3. A centering algorithm $c : QI^* \to QI$ calculates a vector $\overline{\mathbf{x}} \in QI$ that is supposed to represent the elements of a sequence $\mathbf{x}_1, \mathbf{x}_2, \dots \in QI^*$. For the case of identical vectors we assume that $c(\mathbf{x}, \mathbf{x}, \dots) = \mathbf{x}$.

To evaluate the quality of an anonymization algorithm a metric $d(\mathbf{x}, \mathbf{x}')$ is used on the set QI. We extend d to a metric for two sequences $\mathcal{X}_{QI}, \mathcal{X}'_{QI}$ of equal length with the help of a function $f : \mathbb{R}_+^n \to \mathbb{R}_+$ by $d(\mathcal{X}_{QI}, \mathcal{X}'_{QI}) = f(d(\mathbf{x}_1, \mathbf{x}'_1), \dots, d(\mathbf{x}_n, \mathbf{x}'_n))$. The aggregation function f could, for example, be any ℓ_p-norm for $1 \leq p \leq \infty$.

Definition 1.4. The anonymization distortion of an algorithm μ applied to a database $\mathcal{X}$ is defined by $D_\mu(\mathcal{X}) := d(\mathcal{X}_{QI}, \mu(\mathcal{X})_{QI})$.

Scaling the values of a database $\mathcal{X}$ by a factor $\alpha > 1$ will increase the anonymization distortion, too, except for trivial metrics. Thus, distortion should be measured relative to the data expansion in $\mathcal{X}$ which will be called diversity.

The diversity $\Delta(\mathcal{X}) \in \mathbb{R}_+$ should fulfill the condition: $\Delta(\mathcal{X}) = 0$ iff $\mathcal{X}$ is QI-uniform. For example, $\Delta(\mathcal{X})$ could be the sum of pairwise distances $\sum_{1 \le i < i' \le n} d(\mathbf{x}_i, \mathbf{x}_{i'})$ denoted by $\Delta_{\mathtt{PD}}$. Alternatively, one could use the sum of distances to a center of the whole sequence given by $\Delta_c := \sum_i d(\mathbf{x}_i, c(\mathcal{X}_{QI}))$. Then the information loss is defined as the quotient of anonymization distortion and diversity.

Definition 1.5. The information loss $L_\mu(\mathcal{X})$ when applying an anonymization algorithm μ to a non-QI-uniform database $\mathcal{X}$ is defined as $L_\mu(\mathcal{X}) := D_\mu(\mathcal{X})/\Delta(\mathcal{X})$.

The anonymization technique considered in this paper is called *microaggregation* [D09]. Given k it creates a *k-clustering* of a sequence $\mathcal{X}$ that is described by a partition $\mathcal{G} := G_1 \cup G_2 \cup \cdots \cup G_t$ of the indices of the elements of $\mathcal{X}$ into groups G_ℓ such that $|G_\ell| \ge k$ for every ℓ. Let us denote the elements of a group G_ℓ by $\ell_1, \ldots, \ell_{|G_\ell|}$. We call a k-clustering *strict* if each group contains exactly k elements except at most one (in case the size of $\mathcal{X}$ is not a multiple of k).

After the partitioning for every group G_ℓ a representative vector $\overline{\mathbf{x}}_\ell = c(\mathbf{x}_{\ell_1}, \ldots, \mathbf{x}_{\ell_{|G_\ell|}})$ is created by applying a centering algorithm c. In the anonymized output $\hat{X}$ each vector $\mathbf{x}$ belonging to group G_ℓ is replaced by its corresponding group representative $\overline{\mathbf{x}}_\ell$. Obviously, microaggregation achieves k-anonymity.

For microaggregation there are specific choices for the functions c, d, f, Δ which are commonly used for real-valued quasi-identifiers [DMS06, DM02a, DSS08, LM05, SMD06]. The *squared euclidean distance* of two vectors $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^m$ is defined as $d_{\mathtt{SSE}}(\mathbf{x}, \mathbf{x}') := \sum_j (x_j - x'_j)^2$. Now if f is chosen as the sum operator, for two sequences $\mathcal{X}, \hat{\mathcal{X}}$ the value $d_{\mathtt{SSE}}(\mathcal{X}, \hat{\mathcal{X}})$ is called the *sum of squared errors*. This defines the anonymization distortion

$$D_\mu^{\mathtt{SSE}}(\mathcal{X}) \;:=\; \sum_{i=1}^{n} d_{\mathtt{SSE}}(\mathbf{x}_i, \mu(\mathcal{X})_i) \;.$$

For a k-clustering with groups $G_1, \ldots, G_t$ this can be rewritten as $\sum_{\ell=1}^{t} d_{\mathtt{SSE}}(G_\ell)$ where $d_{\mathtt{SSE}}(G_\ell) := \sum_{i \in G_\ell} d_{\mathtt{SSE}}(\mathbf{x}_i, \overline{\mathbf{x}}_\ell)$. In this case it can be shown

Lemma 1.6. If the representative $\overline{\mathbf{x}}_\ell$ of a group G_ℓ consisting of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_{|G_\ell|}$ is chosen as $c_{\mathtt{MEAN}}(\mathbf{x}_1, \ldots, \mathbf{x}_{|G_\ell|})$ then $d_{\mathtt{SSE}}(G_\ell)$ is minimized.

This motivates to measure the diversity of a database $\mathcal{X}$ by considering it as a single group and to measure the individual SSE-distances to a center $\bar{\mathbf{x}}$ for the whole group that is computed by $\bar{\mathbf{x}} := c_{\mathtt{MEAN}}(\mathbf{x}_1, \ldots, \mathbf{x}_n)$. Thus, we define $\Delta^{\mathtt{SSE}}(\mathcal{X}) := \sum_{i=1}^{n} d_{\mathtt{SSE}}(\mathbf{x}_i, c_{\mathtt{MEAN}}(\mathbf{x}_1, \ldots, \mathbf{x}_n))$. For this setting it can be shown:

Lemma 1.7. $\sum_{\ell=1}^{t} d_{\mathrm{SSE}}(G_\ell) \leq \Delta^{\mathrm{SSE}}(\mathcal{X})$ for any partition of $\mathcal{X}$ into groups G_ℓ.

As a consequence, the information loss $L_\mu^{\mathrm{SSE}}(\mathcal{X}) := D_\mu^{\mathrm{SSE}}(\mathcal{X})/\Delta^{\mathrm{SSE}}(\mathcal{X})$ is normalized to the interval $[0, 1]$ for every microaggregation algorithm μ.

The obvious optimization problem for microaggregation is to find a k-clustering into groups G_ℓ that minimizes $\sum_{\ell=1}^{t} d_{\mathrm{SSE}}(G_\ell)$. It has been observed that there always exists an optimal clustering with group sizes between k and $2k - 1$ [DM02a]. Still, finding an optimal clustering remains a computationally difficult problem.

Theorem 1.8. [OD01] Optimal microaggregation for $m \geq 2$ and $k = 3$ is NP-hard for the metric d_{SSE}.

This claim has also be made for $k > 3$ without giving a proof. For $k = 2$ finding an optimal strict clustering can be reduced to the weighted matching problem and thus is efficiently solvable. In the nonstrict case, in an optimal clustering every group can consist of 2 or 3 elements and its computational complexity seems to be open. It is also unclear how well this problem can be approximated. Therefore, good heuristics for arbitrary k are of interest.

The rest of this paper is structured as follows. In section 2 we discuss the standard microaggregation heuristic `MDAV` and two important variants of it. Next in section 3 we present a new strategy for k-clustering called `MDAV`* . This section also contains a concrete example of a simple instance where `MDAV`* outperforms the other algorithms by far. Its complexity is analyzed in section 4. Experimental results on established benchmark databases are presented in section 5 that illustrate the improvements achieved.

2 MDAV

The two most common microaggregation algorithms for k-anonymity are `MDAV` [DM02a][DT05][D09] and the more recent PCL [RFP13]. `MDAV` (maximum distance to average vector) relies on a greedy nearest-neighbor aggregation technique and generates a strict k-clustering, whereas PCL uses a modification of the Lloyd algorithm for aggregation. PCL achieves lower information losses than `MDAV` on synthetical as well as on standard data sets, but this comes at the cost of a substantially longer running time [RFP13]. There are several variations of `MDAV` that differ slightly in time complexity and information loss obtained. We use the `MDAV` specification from [D09].

[SMD06] applies several simplifications and improvements to `MDAV`. Instead of forming two groups at extremal regions simultaneously only a single group is constructed at a time, the global centroid is not recomputed each time, and leftovers at the end are assigned to their closest groups instead of forming a new group out of them. This variant denoted by `MDAV`$^+$ is formally defined below.

Algorithm MDAV$^+$

1. Compute the centroid $\overline{\mathbf{x}}$ of the input dataset $\mathcal{X}$.
2. Select an unassigned record $\mathbf{x}_r$ furthest away from $\overline{\mathbf{x}}$.
3. Form a group around $\mathbf{x}_r$ consisting of $\mathbf{x}_r$ and its $k-1$ closest unassigned neighbors (these elements are now assigned).
4. If there are at least k unassigned records left go back to step 2, otherwise put each not yet assigned element into its closest group.

A significant improvement of MDAV is V-MDAV (Variable group size MDAV) that can generate nonstrict k-clusterings. If a region contains more than k elements, MDAV splits it to obtain groups of fixed size k. V-MDAV solves this problem by considering the inclusion of additional records to newly formed groups.

Let U be the index set of all unassigned records and G be the most recently established group. If the size of G is smaller than $2k-1$ we select a pair of elements $(\mathbf{x}_{\bar{i}}, \mathbf{x}_{\bar{j}})$ with $\bar{i} \in U$ and $\bar{j} \in G$ that minimizes $d_{\mathrm{SSE}}(\mathbf{x}_i, \mathbf{x}_j)$ over all $(i,j) \in U \times G$. Denote this value by $d_{in} := d_{\mathrm{SSE}}(\mathbf{x}_{\bar{i}}, \mathbf{x}_{\bar{j}})$. It is compared to the distance of $\mathbf{x}_{\bar{i}}$ to the closest unassigned element

$$d_{out} := \min_{i \in U - \bar{i}} d_{\mathrm{SSE}}(\mathbf{x}_{\bar{i}}, \mathbf{x}_i).$$

To decide, how to handle $\mathbf{x}_{\bar{i}}$ V-MDAV uses a gain factor γ. $\mathbf{x}_{\bar{i}}$ is added to G iff $d_{in} < \gamma \cdot d_{out}$, otherwise a new group is established around $\mathbf{x}_{\bar{i}}$. For $\gamma = 0$ V-MDAV equals MDAV$^+$. With $0 < \gamma \leq 1$, $\mathbf{x}_{\bar{i}}$ can extend G only if it is closer to G than to its closest unassigned neighbor. For $\gamma > 1$ V-MDAV favors to assign $\mathbf{x}_{\bar{i}}$ to G even if there might be a closer unassigned neighbor. The last decision seems only reasonable for $k > 2$. As stated in [SMD06] it is not clear how an optimal choice of γ should look like. The authors recommend $\gamma = 0.2$ for scattered data sets and $\gamma = 1.1$ for grouped data sets.

3 MDAV*

The main novelty of the heuristic MDAV* is to take into account the effects on nearby records when the extension of a group has to be decided and to handle group extension before creating a new group instead of after the creation. When assigning elements to groups we consider the additional costs per element (marginal costs) a decision would cause and (greedily) select an optimal one.

Before we go into details, consider the simple example of a database $\mathcal{X}$ with a single quasi identifier attribute. It contains 11 records depicted by values $\mathbf{x} \in \{1, 2, 3, 5, 6, 19, 20, 21, 98, 99, 100\}$ of this attribute. The centroid of $\mathcal{X}$ is 34 resulting in diversity $\Delta^{\mathrm{SSE}} = 17966$. For the anonymity parameter we choose $k = 3$.

To construct a k-clustering of $\mathcal{X}$ the heuristic V-MDAV starts with a group G_1 containing the values 98, 99 and 100. Because there are no unassigned elements near that group G_1 does not get expanded. Then the group G_2 with elements 1, 2 and 3 is created – similarly this group will not be expanded because $\mathbf{x}_{\bar{i}} = 5$ is chosen, which results in $d_{in} = d(3,5) = 4$ and $d_{out} = d(5,6) = 1$, thus 5 is not included in G_2. The third group G_3 is created out of 5 and contains 5, 6 and 19. Finally, the elements 20 ($d_{in} = d(19,20) = 1, d_{out} = d(20,21) = 1$) and 21 ($d_{in} = d(20,21), d_{out} = \infty$) join this group. The centroids of these groups are 2, 14.2 and 99 yielding an anonymization distortion of

$$D^{\mathrm{SSE}}_{\text{V-MDAV}} = d_{\mathrm{SSE}}(G_1) + d_{\mathrm{SSE}}(G_2) + d_{\mathrm{SSE}}(G_3) = 2 + 2 + 254.8 = 258.8$$

and information loss $L^{\mathrm{SSE}}_{\text{V-MDAV}} \approx 1.44\%$.

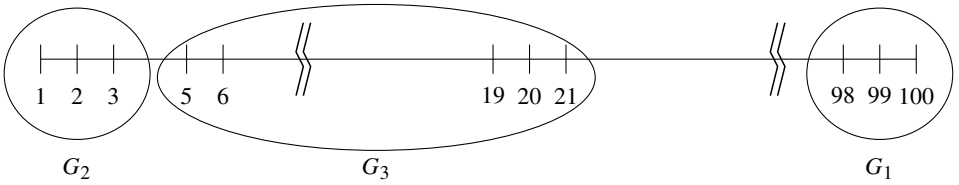

Fig. 1: 3-Clustering generated by V-MDAV

Creation of the group G_3 with 5 elements from 5 to 21 seems to be a bad decision. It is caused by the fact that 5 has a close neighbor 6 to open a new group. However, this should not be considered as reason enough not to include 5 in the already established close group G_2. It would be better to compare the consequences of including 5 in G_2 to building a new group out of 5. This fact brings us back to MDAV*.

The state of MDAV* can be described by a partitioning $\mathcal{G} := G_1 \cup \cdots \cup G_t \cup U$, which consists of t disjoint groups G_ℓ of size at least k and an additional group U containing all records which are still unassigned. Given the state of the algorithm we define the neighborhood $N_q(\mathbf{x}, G) \subseteq G$ as the subset of a group G, containing the indices of $\mathbf{x}$ and its q closest neighbors. Furthermore, $\mathrm{clos}(\mathbf{x}) \in \mathcal{G} \setminus \{U\}$ denotes a group G that is closest to $\mathbf{x}$. If an element $\mathbf{x}$ is included in a group G, we write $G + \mathbf{x}$, if it is removed $G - \mathbf{x}$.

After the choice of a new cluster origin $\mathbf{x}_r \in U$, MDAV* considers two options. The first one is to build a new group $N_{k-1}(\mathbf{x}_r, U)$ as usual. The second one is to extend $\mathrm{clos}(\mathbf{x}_r)$ by $\mathbf{x}_r$ in which case the group $N_{k-1}(\mathbf{x}_r, U)$ cannot be built. Instead the rest of $N_{k-1}(\mathbf{x}_r, U)$ has to be assigned somehow differently. For this, we take the closest neighbor $\mathbf{y} \in U$ of $\mathbf{x}_r$ and consider establishing a new group around $\mathbf{y}$. The underlying decision rule considers the marginal costs in both cases. Still, this is only an estimate of a best possible usage of $\mathbf{x}_r$ because we do not know whether $\mathbf{y}$ should ever be chosen as the origin of a new group.

The costs divided by the number k of elements for creating a new group $N_{k-1}(\mathbf{x}_r, U)$ out of record $\mathbf{x}_r$ are

$$\frac{d_{\mathrm{SSE}}(N_{k-1}(\mathbf{x}_r, U))}{k} \tag{1}$$

while the costs per element of extending the group $\text{clos}(\mathbf{x}_r)$ by $\mathbf{x}_r$ and establishing a new group around $\mathbf{y}$ (now assigning $k+1$ elements) are

$$\frac{d_{\text{SSE}}(\text{clos}(\mathbf{x}_r) + \mathbf{x}_r) - d_{\text{SSE}}(\text{clos}(\mathbf{x}_r)) + d_{\text{SSE}}(N_{k-1}(\mathbf{y}, U - \mathbf{x}_r))}{k+1}. \tag{2}$$

Algorithm MDAV*

1. Compute the centroid $\overline{\mathbf{x}}$ of the input dataset $\mathcal{X}$ and initialize U with $\mathcal{X}$.
2. Select $\mathbf{x}_r \in U$ furthest away from $\overline{\mathbf{x}}$.
3. Compute $N_{k-1}(\mathbf{x}_r, U)$ and the group $\text{clos}(\mathbf{x_r})$.
4. Based on the marginal costs (1) and (2) choose between
   • extending $\text{clos}(\mathbf{x_r})$ by $\mathbf{x}_r$ and removing $\mathbf{x}_r$ from U versus
   • establishing the new group $N_{k-1}(\mathbf{x}_r, U)$ and removing these elements from U.
5. If $|U| \geq k$ go back to step 2,
   otherwise assign each $\mathbf{x} \in U$ to $\text{clos}(\mathbf{x})$.

We are now able to describe how MDAV* handles the situation from above. After creating group G_1 and G_2 as V-MDAV does, considering the next element 5 it is included in G_1, because the costs for creating a new group $N_2(\mathbf{x}_r, U)$ (containing $5, 6$ and 19) are $40.\overline{6}$ whereas the costs for expanding the group G_2 are 32.19. Also 6 is included in G_2, because the costs for a new group are $40.\overline{6}$ and costs for expanding G_2 are only 2.6125. The centroid of G_2 becomes 3.4 and results in a distortion

$$D^{\text{SSE}}_{\text{MDAV}^*} = d_{\text{SSE}}(G_1) + d_{\text{SSE}}(G_2) + d_{\text{SSE}}(G_3) = 2 + 17.2 + 2 = 21.2$$

and information loss $L^{\text{SSE}}_{\text{MDAV}^*} \approx 0.118\%$. Now, comparing $L^{\text{SSE}}_{\text{V-MDAV}}$ and $L^{\text{SSE}}_{\text{MDAV}^*}$ we see that for this instance the information loss of V-MDAV is more than 12 times larger than that of MDAV*.

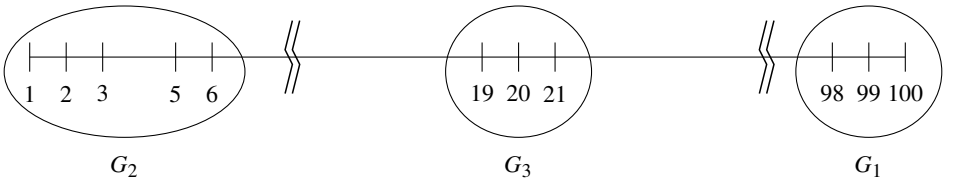

Fig. 2: 3-Clustering generated by MDAV*

An important part missing so far is the calculation of $\text{clos}(\mathbf{x}_r)$. In V-MDAV the distance between a group G and a record $\mathbf{x}$ is measured by the distance between $\mathbf{x}$ and the record from G closest to $\mathbf{x}$. Our goal is to increase d_{SSE} of a group as little as possible when $\mathbf{x}$ is included. However, the increase of group SSE depends on the group's size. This implies that this shortcut may deliver a non-optimal record when searching for the best group to extend with a fixed record.

Fact 3.1. The distance measure between a group and a record used by V-MDAV depends on the group size and is not optimal in general.

To address this problem, MDAV* chooses clos($\mathbf{x}$) by looking directly at the growth of d_{SSE} the extension would cause:

$$\mathrm{clos}(\mathbf{x}) := \operatorname*{arg\,min}_{\mathcal{G}\setminus\{U\}} d_{\mathrm{SSE}}(G + \mathbf{x}) - d_{\mathrm{SSE}}(G) \ .$$

4 Complexity analysis

MDAV$^+$ requires $\lfloor n/k \rfloor$ rounds to create that many groups of size k each except possibly the last one. Computing $\mathbf{x}_r$ and its $(k-1)$-neighborhood can be done in time $O(k\ n)$. This gives a total time complexity of $O(n^2)$.

V-MDAV and MDAV* may run for almost n rounds if the extension of a group by a single element happens often. d_{in} and d_{out} can be computed in $O(k\ n)$ steps which results in an upper time bound $O(k\ n^2)$ for V-MDAV.

To compute clos($\mathbf{x}$) for at most n/k groups their distortion has to be recomputed which requires $O(k)$ steps per group. For the $(k-1)$-neighborhoods time $O(k\ n)$ suffices as in MDAV$^+$. Again this gives the bound $O(k\ n^2)$ for MDAV*.

If we consider k as a constant then the asymptotic growth of the time complexity of these three algorithms is quadratic with respect to the size of the database. This can even be lowered if we order the elements in U with respect to their distance to the centroid and in each round perform a bounded search for closest neighbors.

It remains to compare the performance of these heuristics with respect to the information loss. Since the optimization problem has shown to be hard it will be difficult to compute the minimum possible distortion achievable by clustering. Thus, there is little hope to derive performance bounds analytically in general. To get some insight in how the heuristics perform in practice we have conducted a bunch of experiments.

5 Information loss – experimental results

The information loss has been estimated on different kinds of data. On the one hand, three benchmark data sets (CENSUS, TARRAGONA and EIA) from the CASC project [DM02b] have been used. CENSUS contains 13 numerical attributes and 1080 records. It was created using the Data Extraction System of the U.S. Bureau of Census in 2000. TARRAGONA contains 13 numerical attributes and 834 records. It contains data of the Spanish region Tarragona from 1995. The EIA data set consists of 15 attributes and 4092 records. As in

[DMS06] we only use a subset of 11 attributes precisely 1 and 6 to 15 and ignore categorical data in attribute 2 and 3 as well as attributes with small width in attributes 4 and 5.

We have also tested the information loss on synthetic data. For this process the uniform data set SimU and the grouped data set SimC as in [DMS06] have been created. SimU contains 1000 records with 10 independent numerical attributes which values are chosen uniformly at random. SimC contains clustered data. First 100 cluster centers with 10 attributes are chosen like in SimU. Then for each cluster a number in the interval $[4 \ldots 21]$ is randomly chosen as its size and the cluster is filled with that many elements differing from the cluster center by at most 0.5% in every attribute dimension. Finally $x/3$ independent records are added, where x is the number of records created so far. For uniform as for grouped data we have created 25 data sets and taken the average information loss as final result for all algorithms.

5.1 Test methodology

In all test cases we have standardized the data prior to anonymization. By adjusting the mean value to 0 and the variance to 1 for every attribute the influence by the size of numbers (some attributes have a small range, others a much larger one) has been eliminated. As a consequence no attribute looks more important than others to an anonymization algorithm. L_μ^{SSE} is used on the standardized database $\mathcal{X}$ as information loss measure. The numerical values shown in the tables have been multiplied by 100, thus percentages are listed (the same format as in previous publications). Thus, our results are directly comparable to the results e.g. in [DMS06] or [SMD06]. All tests have been conducted for $k \in \{3, 4, 5, 7, 10\}$. `V-MDAV` has been tested with gain factors γ between 0.0 or 2.0 in steps of 0.1. Only the best result is shown here (see table 4 for the values used in every test case).

5.2 Results

The results are listed in table 1 and table 2 as well as in graphical form in figure 3 to figure 5. As summarized in table 3a) there are only 3 out of 25 test cases in which $\texttt{MDAV}^+$ is slightly better than $\texttt{MDAV}^*$. In all other cases the information loss inflicted by $\texttt{MDAV}^*$ is between 0.9 and 45.4 percent lower than the one of $\texttt{MDAV}^+$ on the same data. On average over the 25 test cases shown the information loss of $\texttt{MDAV}^*$ is about 7.5% lower.

In table 3b) $\texttt{MDAV}^*$ is compared to `V-MDAV`. Because `V-MDAV` can behave like $\texttt{MDAV}^+$ (setting $\gamma = 0$) it can never be worse than $\texttt{MDAV}^+$ if for every instance the (unknown) optimal scaling factor is used. The results show that there are scenarios in which `V-MDAV` takes profit from this. However, in most cases $\texttt{MDAV}^*$ has an even lower information loss than `V-MDAV`. This clearly shows the further improvement achieved by $\texttt{MDAV}^*$.

Another interesting question is how the information loss increases with k. For our data sets this has a significant impact. In figure 6 the impact of k for the SimC test in the range $k \in \{2, 3, \ldots, 100\}$ is shown. The graphs for other test cases look similar and are omitted here.

	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
MDAV	5.677	7.537	9.056	11.608	14.186
MDAV$^+$	5.662	7.514	9.007	11.657	14.073
V-MDAV	5.662	7.514	8.978	11.586	14.043
MDAV*	5.782	7.433	8.809	11.369	14.003

(a) CENSUS

	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
MDAV	16.922	19.540	22.459	27.525	33.195
MDAV$^+$	16.951	19.767	22.872	28.255	33.254
V-MDAV	15.849	19.695	22.872	28.249	33.251
MDAV*	16.143	19.189	22.250	28.399	34.743

(b) TARRAGONA

	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
MDAV	0.483	0.672	1.668	2.187	3.846
MDAV$^+$	0.488	0.673	1.775	2.211	3.547
V-MDAV	0.465	0.673	1.056	2.211	2.794
MDAV*	0.449	0.617	0.911	2.032	2.633

(c) EIA

Tab. 1: Information loss on standard test data sets given in percentages ($100 \cdot L_\mu^{\mathrm{SSE}}$)

6 Future work

Can the clustering problem be solved by an approximation algorithm with a guaranteed approximation rate for the information loss? The only result known to us is an algorithm in [DSS08] with a quite high rate of $O(k^3)$. It should be possible to improve this bound significantly. The simple 1-dimensional example given above illustrates that in the worst case MDAV* might be much better than V-MDAV. It would be interesting to prove a bound on the approximation ratio of MDAV* or a further improved strategy.

The property k-anonymity has been extended to stronger privacy requirements called ℓ-diversity and t-closeness. However, these properties seem to induce even higher information loss and algorithmic solutions are significantly more difficult to analyze in a rigorous way. Can one establish any performance guarantees for these settings?

	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
MDAV	18.181	23.656	28.072	34.671	40.826
MDAV$^+$	18.026	23.618	27.897	34.043	40.625
V-MDAV	17.993	23.551	27.803	34.006	40.427
MDAV*	17.756	23.095	27.280	33.371	39.643

(a) SimU

	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
MDAV	6.957	9.294	11.246	14.511	18.436
MDAV$^+$	6.856	9.233	11.045	14.114	18.086
V-MDAV	6.192	8.355	10.155	13.142	16.973
MDAV*	5.999	8.163	9.659	12.315	15.683

(b) SimC

Tab. 2: Information loss on synthetic test data sets given in percentages ($100 \cdot L_\mu^{\mathrm{SSE}}$)

	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
CENSUS	+1.8%	−1.4%	−2.7%	−2.1%	−1.3%
TARRAGONA	−4.6%	−1.8%	−0.9%	+3.2%	+4.7%
EIA	−7.0%	−8.1%	−45.4%	−7.1%	−31.5%
SimU	−2.3%	−2.4%	−2.8%	−3.7%	−2.9%
SimC	−13.8%	−12.2%	−14.1%	−15.1%	−14.9%

(a) MDAV* versus MDAV$^+$

	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
CENSUS	+2.11%	−1.08%	−1.89%	−1.87%	−0.29%
TARRAGONA	+1.85%	−2.57%	−2.72%	+0.53%	+4.49%
EIA	−3.40%	−8.23%	−13.71%	−8.09%	−5.77%
SimU	−1.32%	−1.93%	−1.88%	−1.86%	−1.94%
SimC	−3.11%	−2.29%	−4.88%	−6.30%	−7.60%

(b) MDAV* versus V-MDAV

Tab. 3: Percental information loss difference of MDAV* compared to MDAV (a) and V-MDAV (b) Note that negative numbers show an improvement.

	$k = 3$	$k = 4$	$k = 5$	$k = 7$	$k = 10$
CENSUS	$\gamma = 0.0$	$\gamma = 0.0$	$\gamma = 0.2$	$\gamma = 0.1$	$\gamma = 0.2$
TARRAGONA	$\gamma = 0.3$	$\gamma = 0.3$	$\gamma = 0.0$	$\gamma = 0.6$	$\gamma = 0.3$
EIA	$\gamma = 0.6$	$\gamma = 0.0$	$\gamma = 0.4$	$\gamma = 0.0$	$\gamma = 1.3$
SimU	$\gamma = 0.176$	$\gamma = 0.208$	$\gamma = 0.232$	$\gamma = 0.108$	$\gamma = 0.244$
SimC	$\gamma = 0.368$	$\gamma = 0.456$	$\gamma = 0.528$	$\gamma = 0.660$	$\gamma = 0.776$

Tab. 4: Optimal gain factors used for V-MDAV in the experiments. For SimU and SimC the arithmetic mean of the optimal γ for each of the 25 sets is shown.

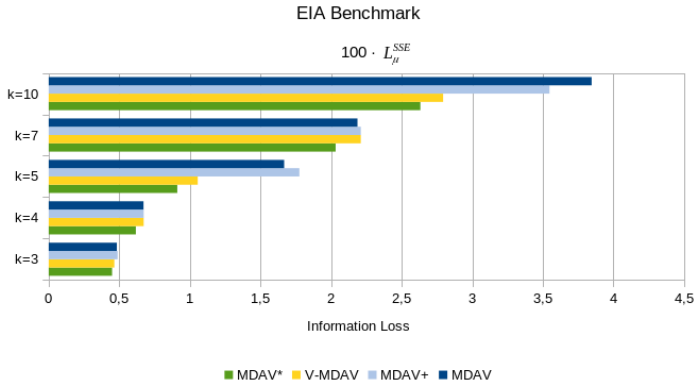

Fig. 3: Information loss for EIA benchmark

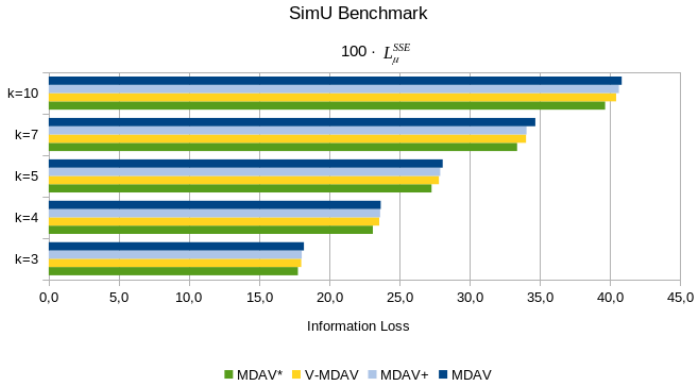

Fig. 4: Information loss for SimU benchmark

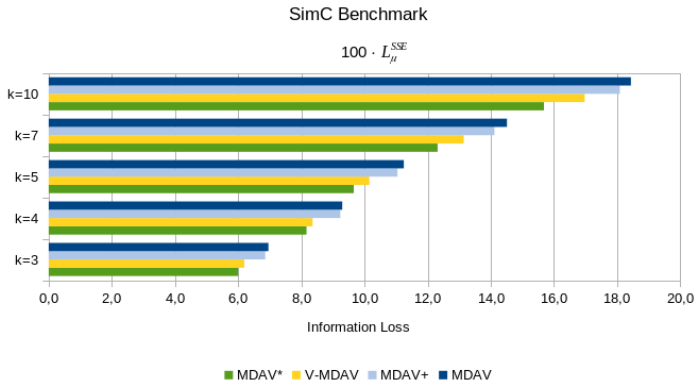

Fig. 5: Information loss for SimC benchmark

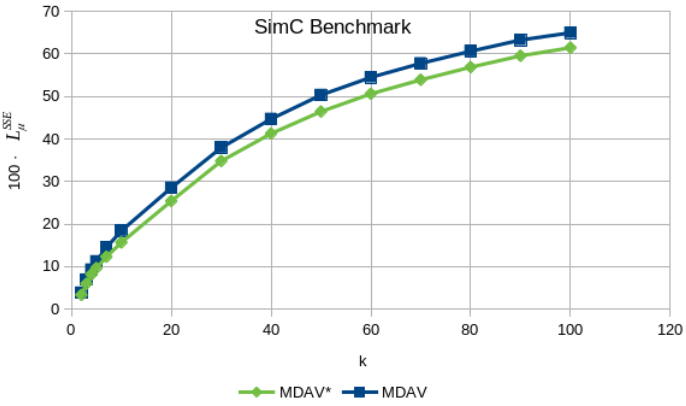

Fig. 6: Information loss tendency for different k

References

[D09] Domingo-Ferrer, J.: Encyclopedia of Database Systems. Springer US, chapter Microaggregation, pp. 1736–1737, 2009.

[DM02a] Domingo-Ferrer, J.; Mateo-Sanz, J. M.: Practical data-oriented microaggregation for statistical disclosure control. IEEE Transactions on Knowledge and data Engineering, 14(1):189–201, 2002.

[DM02b] Domingo-Ferrer, J.; Mateo-Sanz, J. M.: , Reference data sets to test and compare sdc methods for protection of numerical microdata. `http://neon.vb.cbs.nl/casc`, 2002.

[DSS08] Domingo-Ferrer, J.; Sebé, F.; Solanas, A.: A polynomial-time approximation to optimal multivariate microaggregation. Computers & Mathematics with Applications, 55(4):714–732, 2008.

[DT05] Domingo-Ferrer, J.; Torra, V.: Ordinal, continuous and heterogeneous k-anonymity through microaggregation. Data Mining and Knowledge Discovery, 11(2):195–212, 2005.

[DMS06] Domingo-Ferrer, J.; Martínez-Ballesté, A.; Mateo-Sanz, J. M.; Sebé, F.: Efficient multivariate data-oriented microaggregation. The VLDB Journal—The International Journal on Very Large Data Bases, 15(4):355–369, 2006.

[LM05] Laszlo, M.; Mukherjee, S.: Minimum spanning tree partitioning algorithm for microaggregation. IEEE Transactions on Knowledge and Data Engineering, 17(7):902–911, 2005.

[OD01] Oganian, A.; Domingo-Ferrer, J.: On the complexity of optimal microaggregation for statistical disclosure control. Statistical Journal of the United Nations Economic Commission for Europe, 18(4):345–353, 2001.

[RFP13] Rebollo-Monedero, D.; Forné, J.; Pallarès, E.; Parra-Arnau, J.: A modification of the Lloyd algorithm for k-anonymous quantization. Information Sciences, 222:185–202, 2013.

[SMD06] Solanas, A.; Martinez-Balleste, A.; Domingo-Ferrer, J.: V-MDAV: a multivariate microaggregation with variable group size. In: 17th COMPSTAT Symposium of the IASC, Rome. pp. 917–925, 2006.

[S02] Sweeney, L.: k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(05):557–570, 2002.