Journal of Ambient Intelligence and Humanized Computing manuscript No.
(will be inserted by the editor)

Hybrid Microaggregation for Privacy-Preserving Data Mining

Balkis Abidi · **Sadok Ben Yahia** · **Charith Perera**

Received: date / Accepted: date

Abstract *k*-Anonymity by microaggregation is one of the most commonly used anonymization techniques. This success is owe to the achievement of a worth of interest trade-off between information loss and identity disclosure risk. However, this method may have some drawbacks. On the disclosure limitation side, there is a lack of protection against attribute disclosure. On the data utility side, dealing with a real datasets is a challenging task to achieve. Indeed, the latter are characterized by their large number of attributes and the presence of noisy data, such that outliers or, even, data with missing values. Generating an anonymous individual data useful for data mining tasks, while decreasing the influence of noisy data is a compelling task to achieve.
In this paper, we introduce a new microaggregation method, called HM-PFSOM, based on fuzzy possibilistic clustering. Our proposed method operates through an hybrid manner. This means that the anonymization process is applied per block of similar data. Thus, we can help to decrease the information loss during the anonymization process. The HM-PFSOM approach proposes to study the distribution of confidential attributes within each sub-dataset. Then, according to the latter distribution, the privacy parameter k is determined, in such a way to preserve the diversity of confidential attributes within the anonymized microdata. This allows to decrease the disclosure risk of confidential information.

Balkis Abidi
LIPAH, Faculty of Sciences of Tunis, University of El-Manar, Tunisia

Sadok Ben Yahia
LIPAH, Faculty of Sciences of Tunis, University of El-Manar, Tunisia

Charith Perera
School of Computing Science, Newcastle University, United Kingdom

Keywords Hybrid micoaggregation · Information loss · Identity disclosure risk · Attribute disclosure risk · Fuzzy possibilistic clustering

1 Introduction

The ever growing privacy concern has been a major obstacle for individual data analysis. In fact, many situations require that governmental, public and private organizations share and release their specific data (Chui et al 2014). These latter, generally, reflect our everyday life activities, *e.g.* credit card transactions, activities on the web, phone calls, widespread diseases, *etc*. Publishing and releasing such type of data can provide benefits to researchers and decision makers, owe to their flexibility and availability of detailed information (Teplitzky 2014). For example, healthcare organizations collect and analyze medical data for the discovery of new drugs and therapies (Rider and Chawla 2013). Retail companies need information about customers, in order to identify customer purchases behaviours, discover customer shopping patterns and trends, and thus improve the quality of customer services (Peersman 2014). Banks and financial institutions also collect financial data to predict credit fraud, evaluate risk and perform trend analysis (Bennardo et al 2015). As for telecommunication companies, they maintain a great deal of call detail data, which describe the calls that traverse telecommunication networks (Chittaranjan et al 2013). Such data can be useful to identify vulnerabilities of networks. Social networks are undoubtedly the most extreme example of data valorisation. The deal is to provide users a free social media platform to entertain, in return collect all kinds of information describing the users' relationships, interests, apps in use, and also religion or political opinions (Johnson et al 2012). Such information are used to sell advertising and insights based on their profiles. The Facebook-Cambridge An-

alytica data scandal is the prime example of how personal data could be disclosed, where the collection of personally identifiable information of up to 87 million Facebook users was allegedly used to attempt to influence voter opinion on behalf of politicians who hired them (Solon 2018).

However, individual data may contain confidential information. Thus, collecting, analyzing and sharing such data, raises threat to individual privacy. Data de-identification, *i.e.* hiding explicit *identifiers*, is considered of paramount importance to avoid sensitive information from being disclosed. Such process involves removing any information which is able to *uniquely* identify an individual, *e.g.* name, SSN, *etc* (Garfinkel 2015). Nevertheless, the latter process could not guarantee efficient security. Indeed, it was shown that is possible to manipulate de-identified datasets and recover personal information, through data linkage techniques (Ohm 2010). It is worth mentioning that disclosure risk can be classified into two categories, namely (Hundepool et al 2012):

- *Identity disclosure*: The intruder is able to determine the real identity of individual corresponding to a record in the published microdata. Thus, the intruder can associate the confidential information to the re-identified data subject.
- *Attribute disclosure*: Even if identity disclosure does not happen, it may be possible for an intruder to infer some information for a specific individual based on the published microdata.

Therefore, a large number of Privacy-Preserving Data Mining (PPDM) methods have been proposed aiming at ensuring privacy of the respondents, while preserving the statistical utility of the original data (Aggarwal and Yu 2008). The basic idea of this research area is to modify the collected data, subject to be released, in such a way to perform effectively analyses and knowledge discovery tasks without compromising the security of sensitive information contained in the data. Such process leads to reduce the granularity of information, which can cause a loss of data effectiveness. Thus, finding a trade-off between the two conflicting principles, *i.e.* privacy and data utility, is of the utmost importance in PPDM process. Microaggregation (Domingo-Ferrer 2008), is a widely accepted PPDM method for data anonymization. The principle is to de-associate the relationship between the identity of data subjects and their confidential information. Given a security parameter k, the basic idea of microaggregation is to split a dataset into small *groups*, of size at least k. Then, the values of the original data are replaced by those of the cluster's centroid to which they belong to. Thereby, any data is indistinguishable among other $(k-1)$ data. The resulting anonymous dataset fulfils the *k-anonymity* model (Sweeney 2002). Thereby, privacy is ensured by preventing record linkage. Meanwhile, data utility is maintained by gathering records that share the same characteristics. However, generating an *optimal* k-partition while maintaining the homogeneity of data within a fixed size group has been shown a NP-hard problem (Oganian and Domingo-Ferrer 2001). So, the only practical microaggregation methods are based on heuristics. Generally, these methods rely on *refinement* steps, during the partitioning process, which consists in merging or splitting the obtained fixed size groups. However, in real datasets the poor homogeneity within the generated partition can be significant. Thus, its refinement could be costly and does not necessarily converge to the optimal partition. Besides, all microaggregation methods have focused on decreasing the information loss, while maintaining the constraint of the group's size. As consequence, the modified dataset can be exposed to attribute linkage. In fact, k-Anonymity can create groups that leak information due to lack of diversity in the sensitive attribute (Machanavajjhala et al 2007).

This study aims to deal with these shortcomings, by proposing a new algorithm called HM-PFSOM. The proposed algorithm relies on the following assumptions:

- Preventing the identity disclosure risk by requiring that the generated partition should fulfil the k-anonymity property.
- Preventing the attribute disclosure by ensuring the diversity of confidential attributes within each fixed size group of the generated partition.
- Maintaining the data utility of the anonymous microdata by increasing the homogeneity within the fixed size groups, *i.e.* k-partition.

In order to meet the latter requirements, the HM-PFSOM algorithm operates in an *hybrid* manner, *i.e.* per block of data. Indeed, to avoid the risk of gathering data with dissimilar quasi-identifiers in a same group, the HM-PFSOM algorithm splits the microdata into disjoint sub-microdata, through a fuzzy clustering algorithm. The clustering process is applied according to the quasi-identifiers, in order to apply the microaggregation process independently on each sub-microdata, *i.e.* group of data having similar quasi-identifiers. Unlike the standard microaggregation methods, the HM-PFSOM algorithm don't require a *predefined* privacy parameter k, fixed arbitrary, to build the k-anonymous microdata. It proposes to study the distribution of confidential attributes within each sub-microdata. Then, according to the latter distribution, the parameter k is determined, in such a way to preserve the diversity of confidential attributes within the anonymized microdata.

The paper is organized as follows. Section 2 discusses the main PPDM approaches, in particular microaggregation methods. Section 3 introduces the novel **HM-PFSOM** algorithm for **H**ybrid **M**icroaggregation by using the **PFSOM** clustering method. Finally, section 4 evaluates the performance of the proposed approach, through experiments car-

ried out on real-life datasets. Finally, Section 5 sketches our contributions and points out avenues of future work

2 Related work and motivation

In this section we present a succinct introduction to data anonymization and sanitization techniques. We focus in particular on microaggregation model, which is one of the most popular, studied and used PPDM methods. A discussion is also presented, in order to situate our contribution with respect to the reviewed ones of the literature.

2.1 Data anonymization and sanitization techniques

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a de-identified dataset, also called *microdata*, subject to be released. Each input data $x_i \in X$ is a set of m attributes $A = \{a_1, a_2, \ldots, a_m\}$, and $x_i[a_j]$ denotes the value of attribute a_j of the data x_i. We assume that X is a subset of some larger population Ω where each input data represents an individual. The set of attributes A is composed primarily by (Sweeney 2002):

- *Quasi-identifiers* $Qid = \{qid_1, qid_2, \ldots, qid_p\}$, which are a non-sensitive attributes. Nevertheless, if the latter are linked with external information they can *uniquely* identify at least one individual.
- *Confidential attributes* $S = \{s_1, s_2, \ldots, s_q\}$, also called *sensitive attributes*, which their values for any particular individual must be kept secret from users who have no direct access to the original data;

Thus, the set of attributes A can be represented by $Qid \cup S$, *i.e.* $A = \{qid_1, qid_2, \ldots, qid_p\} \cup \{s_1, s_2, \ldots, s_q\}$, where $p + q = m$.

Perturbation of data is a very easy and effective method for protecting the sensitive information of individual data from unauthorised users or hackers. These methods allow the release of the entire microdata, although by masking the sensitive information (Liu et al 2008). This requires to generate a set of random values, which will be used subsequently to hide sensitive information of the released data (Kargupta et al 2005). To maintain privacy and data utility, data perturbation methods aim to replace the original values of data with some artificial values, while maintaining their statistical properties. As the perturbed data records does not match with the original ones, the attacker cannot recover the sensitive information from the perturbed data.
The most widely used techniques, in data perturbation methods, are additive noise and multiplicative stochastic noise (Agrawal and Srikant 2000)(Chen and Liu 2008)(Ciriani et al 2007) (Du and Zhan 2003)(Evfimievski et al 2002)(Mivule 2013). To maintain the effectiveness of data, the generated random values require to comply with the distribution of the original ones. Thus, the distribution of each data dimension is reconstructed independently. This means that any distribution based data mining algorithm works under an implicit assumption to treat each dimension independently (Matwin 2013). However, in many cases, a lot of relevant information for data mining algorithms is hidden in inter-attribute correlations.
Microaggregation technique (Domingo-Ferrer 2008) relaxes the constraint of hiding sensitive attributes with random values. Its principle consists in de-associating the relationship between quasi-identifiers and confidential attributes of individual records. Given a security parameter k, the basic idea of microaggregation is to split a dataset into small *groups*, of size at least k. Then, the quasi-identifiers of the original data are replaced with those of the cluster's centroid to which they belongs to. Thereby, any data is indistinguishable among other $(k-1)$ data. The resulting anonymous dataset fulfils the *k-anonymity* model (Sweeney 2002). Thus, privacy is ensured by preventing record linkage. Meanwhile, data utility is maintained by gathering records sharing the same characteristics of quasi-identifiers.

2.2 The k-anonymous microaggregation model

Normally, microaggregation gathers the closest data in the same fixed size group, in such a way that the respective distances between the data vectors and the corresponding centroids is as small as possible. Thus, the microaggregation generates a protected microdata X' that is similar to the original one, but where data in X' are slightly different from those of X. The optimal k-partition is defined as that it maximizes the within-group homogeneity. The higher the within-group homogeneity, the lower the information loss is. However, finding the optimal cluster configuration has been shown to be a NP-hard problem (Oganian and Domingo-Ferrer 2001). This issue has grasped the interest of the literature and a wealthy number of methods exist.
According to data dimensionality, microaggregation methods can be split into two categories, namely, *univariate* and *multivariate* approaches. The k-partitioning mechanism is the same in all such methods: first, data vectors are sorted in ascending or descending order according to some criterion. Then, groups of successive k vectors are combined. Inside each group, the effect for each variable is to replace the k values taken by the variable with their average. If the total number of data vectors n is not a multiple of k, the last group will contain more than k data vectors.

2.2.1 Univariate microaggregation methods

Univariate microaggregation performs a straightforward *one-dimensional* sorting. To this end, projected methods, also

called *single axis*, are used to summarize the p quasi-identifiers of each data vector into a *single* value (Nin and Torra 2009). The most commonly used methods are *Principal Components Analysis* and the sum of *Zscores* (Nin et al 2008)(Templ 2008). To do so, all attributes are firstly standardized and, for each data vector, the standardized values are added. Vectors are subsequently sorted, *w.r.t.* the scores of the first principal component or by their sum of z-scores. This approach has been shown to be very useful with highly correlated data, *i.e.*, the higher the correlation is, the lower the information loss (Nin et al 2008). However, in real-life datasets, data are not necessarily so highly correlated, which makes these approaches so ineffective.

Another alternative for univariate microaggregation is to apply an anonymization process to each variable independently, *i.e.* data vectors are sorted by the first variable, then groups of k successive values of the first variable are formed and, inside each group, values are replaced by the group average. A similar procedure is repeated for the remainder of variables. This method is referred to as *individual sorting* (Domingo-Ferrer and Mateo-Sanz 2002). Even though, this method usually maintains the data utility, it has a higher disclosure risk, *i.e.* no k-anonymity will be achieved in general. Indeed, by just taking into account the first variable, the k data standing in the same cluster might be assigned to different clusters when all the other variables are considered.

2.2.2 Multivariate microaggregation methods

When the multivariate data are microaggregated without projecting them in one-dimension, this is referred to as a *multivariate microaggregation*. The Maximum Distance to Average Vector (MDAV) algorithm (Domingo-Ferrer and Torra 2005) is the most used for microaggregartion, which operates through an iterative process. Its principle involves computing the centroid of the quasi-identifiers of all input data. The distance of the data to the obtained centroid is used as a sorting criteria. To achieve that, two extreme data, x_r and x_d, relative to the centroid are extracted. Where x_r is the most distant data vector to the centroid, and x_d is the most distant data vector to x_r. Then, two groups are formed with size k around x_r and x_d, respectively. Such process is repeated until all input data vectors of the original microdata are partitioned. Then, the MDAV algorithm generates a partition of fixed size groups having a same cardinality, which is equal to k. Note that, if the number of input data is not divisible by k, the cardinality of one group, generally the last one, ranges between k and $2k-1$. However, the obtained partition of the MDAV algorithm may lack flexibility for adapting the group size constraint to the distribution of the data vectors (Domigo-Ferrer et al 2006). Several microaggregation methods have been proposed to improve the homogeneity within the fixed size groups obtained by the MDAV algorithm (Chang et al 2007) (Domigo-Ferrer et al 2006) (Domingo-Ferrer and Úrsula González-Nicolás 2010) (Lin et al 2010) (Martínez-Ballesté et al 2007) (Solanas et al 2012). Generally these methods add further steps, called *refinement* steps, which consists in merging or splitting the obtained fixed size groups. Thus, yielding a more freedom partition, having groups with different cardinality varying between k and $2k-1$.

However, in real datasets the poor homogeneity within the generated partition can be significant. So, its refinement could be costly and does not necessarily converge to the optimal partition. Besides, all microaggregation methods have focused on the constraint of homogeneity of the k-partition to reduce the information loss. As consequence, the modified dataset can be exposed to an attribute disclosure risk.

2.3 Discussion & Motivation

In our opinion, the major weakness of micoaggregation methods consists in applying the k-partitioning process without studying the distribution of the input data and their correlation. We are convinced that, if there is a step to add, in order to converge to the optimal k-partition, it should be applied *before* the partitioning process. This step should analyze the similarity between the quasi-identifiers of the input data, in order to decide which data should be gathered in a same group. Therewith, we are of the view that to prevent against identity disclosure, the parameter k should be fixed according to the distribution of sensitive information, in order to ensure that the k-anonymous records fulfil a *diversity* of sensitive information and so prevent attribute disclosure.

To meet these issues, we propose a new method, called HM-PFSOM, for hybrid microaggregation, by using PFSOM algorithm for fuzzy possibilistic clustering (Abidi and Ben Yahia 2013). The main idea of the HM-PFSOM algorithm consists in splitting the original dataset into disjoint sub-datasets, in such a way that data within the same sub-dataset must be similar to some extend. In addition, they should be dissimilar to those data in other sub-datasets. Such process enables to avoid the refinement phases used to fine tune the k-anonymous partition, since the partitioning process is performed only on similar data. Indeed, the HM-PFSOM algorithm applies the k-partitioning process independently on each sub-dataset.

On privacy side, the HM-PFSOM algorithm proposes to study the distribution of confidential attributes within each sub-dataset. Then, according to the latter distribution, the privacy parameter k is determined, in such a way to maintain the diversity of confidential attributes within the anonymous microdata.

3 Data anonymisation by hybrid microaggregation

In the following, we introduce a new method, called HMPFSOM, to balance between the above conflicting issues, in order to achieve the tedious optimal partition used to generate an anonymous microdata.

3.1 General principle of the hybrid microaggregation

We propose a new microaggregation method, called hybrid microaggregation, for privacy-preserving data mining. The proposed method, sketched in Algorithm 1, follows the following steps:

1. *Splitting step*: This step aims to split the original microdata into disjoint sub-microdata.
2. *Partitioning step*: This step generates a fixed size partition from each sub-microdata.
3. *Merging step*: In this step, the generated fixed size partitions are used to train a k-anonymous microdata.

Algorithm 1: The general principle of the hybrid microaggregation method

Input: X : The original microdata
Output: X' : The anonymous microdata
Begin
  Split the microdata X into c disjoint sub-microdata
  $X = Xid_1 \cup Xid_2 \cup \ldots \cup Xid_c$.
  Foreach *sub-microdata* $Xid_j \in \{Xid_1, Xid_2, \ldots, Xid_c\}$, $\forall j \in \{1, \ldots, c\}$ **do**
    $X'_j \leftarrow$ Partitioning_process(Xid_j)
  $X' = X'_1 \cup X'_2 \cup \ldots \cup X'_c$
End

The hybrid microaggregation integrates an additional step which consists in discovering the distribution of the quasi-identifiers of all input data. This step aims to decide which data that *can not* belong to a same fixed-size group. Doing so, the hybrid microaggregation starts by splitting the original microdata X into a set of disjoint sub-microdata, *i.e.* $X = \{X_1, X_2, \ldots, X_c\}$, where each sub-microdata gathers similar data. Then, an adaptive partitioning process is applied independently on each sub-microdata $X_i, i = \{1, \ldots, c\}$, to train a k_i-partition. Afterward, the anonymous microdata is obtained from the generated k_i-partition, $i = \{1, \ldots, c\}$.
The adaptive partitioning process should maintain the diversity of confidential attributes within each fixed size group. The latter constraint is paramount importance in the sake of avoiding attribute disclosure risk. To better understand such constraint, let X_{micro}, given in Table 1, be an original microdata. Each input data $x_i \in X_{micro}$ is characterized by a set of two-dimensional quasi-identifiers, *i.e. ZIP code* and *Age*, and one confidential attribute, *i.e. Disease*.

Table 1: A microdata sample X_{micro}

	ZIP code	Age	Disease
x_1	2025	28	Heart Disease
x_2	2022	29	Heart Disease
x_3	2022	25	Viral Infection
x_4	2020	24	Viral Infection
x_5	1012	50	Cancer
x_6	1012	55	Heart Disease
x_7	1013	47	Viral Infection
x_8	1013	49	Viral Infection
x_9	1023	31	Cancer
x_{10}	1022	34	Cancer
x_{11}	1021	35	Cancer
x_{12}	1021	37	Cancer

Table 2: The 4-anonymous microdata of X_{micro}

		ZIP code	Age	Disease
G_1	x_1	**2022**	**27**	Heart Disease
	x_2	**2022**	**27**	Heart Disease
	x_3	**2022**	**27**	Viral Infection
	x_4	**2022**	**27**	Viral Infection
G_2	x_5	**1012**	**50**	Cancer
	x_6	**1012**	**50**	Heart Disease
	x_7	**1012**	**50**	Viral Infection
	x_8	**1012**	**50**	Viral Infection
G_3	x_9	**1021**	**34**	Cancer
	x_{10}	**1021**	**34**	Cancer
	x_{11}	**1021**	**34**	Cancer
	x_{12}	**1021**	**34**	Cancer

Table 2 shows an example of a k-anonymous microdata generated from the original microdata X_{micro}, via a standard microaggregation technique, where the privacy parameter k is set to 3. Note that, the microaggregation is applied in such a way that the obtained anonymous microdata fulfils the k-anonymity property. For any combination of values of quasi-identifiers in the released microdata, there are at least 3 records sharing that combination of values. The fixed size partition is obtained such that each 3 similar data are gathered in same group. The similarity is measured on the basis of the quasi-identifiers attributes.
However, an anonymous microdata, as given in Table 2, can be sensitive to attribute linkage attack, *i.e.* an intruder can discover the confidential information of a given individual. Namely, suppose that the intruder has some knowledge about an individual, for example he is aged 35 years old and he lives in a city where its *ZIP code* is 1022. Thus, by linking these information with the 3-anonymous microdata, the intruder can conclude that the individual in question corresponds to one of the four records having a cancer disease.
To prevent such disclosure, it would be better to ensure, during the partitioning process, the *diversity* of confidential at-

tributes within each fixed size group, as shown in Table 3.

Table 3: The optimal 3-partition of X_{micro}

		ZIP code	Age	Disease
G_1	x_6	1012	55	Heart Disease
	x_5	1012	50	Cancer
	x_8	1013	49	Viral Infection
G_2	x_7	1013	47	Viral Infection
	x_{12}	1021	37	Cancer
	x_{10}	1022	34	Cancer
	x_2	2022	29	Heart Disease
G_3	x_{11}	1021	35	Cancer
	x_1	2025	28	Heart Disease
	x_3	2022	25	Viral Infection
	x_4	2020	24	Viral Infection
	x_9	1023	31	Cancer

In the following, we present in detail the major steps of the proposed HM-PFSOM algorithm for hybrid microaggregation.

3.2 The HM-PFSOM algorithm for hybrid microaggregation

The HM-PFSOM is a new algorithm for hybrid microaggregation aiming to protect microdata from individual identification, that could be identity or attribute disclosure risk. The HM-PFSOM algorithm proceeds by extracting the optimal partition of a given microdata, which will be used to generate the k-anonymous microdata. Thus, the obtained microdata can avoid, or at least decrease, the identity disclosure risk. However, the main originality of the HM-PFSOM algorithm is that it *redefines* the process of extracting the k-partition, by supposing that the cardinality k of each group should not be fixed arbitrary, but it should retain the diversity of confidential attributes.
The pseudo-code of the HM-PFSOM algorithm is sketched by Algorithm 2.

The HM-PFSOM algorithm starts by splitting the original microdata into disjoint sub-microdata (*line* 2), in such a way that data sharing similar characteristic of quasi-identifiers are gathered in the same sub-microdata. To do so, the HM-PFSOM algorithm, relies on fuzzy possibilistic clustering process (Abidi and Ben Yahia 2013). Thereafter, the partitioning process of the microaggregation can be applied independently on each sub-microdata (*lines* $4-17$). Accordingly, the risk of gathering dissimilar data in a same group will be eliminated. Thus, the HM-PFSOM algorithm avoids the refinement steps used to improve the homogeneity of the final k-partition.

Algorithm 2: The HM-PFSOM algorithm

Input:
- X : The original microdata
- k : Privacy parameter

Output: X' : The anonymous microdata

Begin

2 Split the microdata X into c disjoint sub-microdata, according to the quasi-identifiers, *i.e.* $X = Xid_1 \cup Xid_2 \cup \ldots \cup Xid_c$.

3 */* Apply an hybrid microaggregation process */*

4 **Foreach** *sub-microdata* $Xid_j \in \{Xid_1, Xid_2, \ldots, Xid_c\}$, $\forall j \in \{1, \ldots, c\}$ **do**

5 Let $\bar{x}_{id(j)}$ be the cluster centers of the sub-microdata Xid_j.

6 Split the sub-microdata Xid_j into cs_j disjoint clusters, according to the confidential attributes.

7 Let $Cs = \{C_1, C_2, \ldots, C_{cs_j}\}$ be the set of obtained clusters.

8 **While** $|C_i| \geq k, \forall C_i \in Cs$ **do**

9 Extract x_r the most distant record to $\bar{x}_{id(j)}$.

10 */* Form a fixed size group around* x_r **/*

11 Let C_r be the cluster to which belong the data vector x_r.

12 Extract from the cluster C_r the $(k-1)$ closest data vector to x_r

13 Remove the extracted data vector from C_r

14 **Foreach** $C_i \in Cs - \{C_r\}$ **do**

15 Extract the k closest data vector to x_r

16 Remove the extracted data vector from C_i

17 Assign the remaining data to their nearest group.

18 */* Generate an anonymous sub-microdata* X'_j **/*

19 Within each formed group, replace the values of each quasi-identifier attribute with the average value of the attribute over the group.

20 $X' = X'_1 \cup X'_2 \cup \ldots \cup X'_c$

21 **End**

To ensure the diversity of confidential attributes within the k-partition, the HM-PFSOM algorithm studies the distribution of confidential attributes within each sub-microdata $Xid_j \subset X, \forall j \in \{1, \ldots, c\}$. Then, the fixed size groups should be generated in such a way to fulfil the latter distribution. To achieve such purpose, the HM-PFSOM algorithm extracts at first, from each sub-microdata Xid_j, the cs_j disjoint clusters of confidential attributes (*line* 6). The latter clusters are used thereafter for generating fixed size groups. Unlike the standard microaggregation methods, the HM-PFSOM algorithm imposes a condition on the group size. Each group should gather at least cs_j data vectors belonging to different clusters of confidential attributes. To do so, the data vectors of Xid_j are distributed into cs_j disjoint groups, according to their confidential attributes. These data should be close as possible in terms of quasi-identifiers and distant in terms of confidential attributes. In order to respond to such constraint the HM-PFSOM applies an adaptive partitioning process. It computes at first the center of the sub-microdata, noted by $\bar{x}_{id(j)}$ (*line* 5). Then, a data vector x_r is extracted. The lat-

ter corresponds to the most distant data to the centroid $\bar{x}_{id(j)}$ (*line* 9). Let C_r be the cluster to which x_r belongs. The HM-PFSOM algorithm selects firstly the $k-1$ closest data vectors from the clusters C_r (*line* 12). Then, from each cluster C_i, $\forall i \in \{1,\ldots,r-1,r+1,\ldots,c_s\}$, the k closest data vector to x_r are either extracted (*lines* $14-16$). Such process is repeated until all fixed size groups, *i.e.* of cardinality $k \times cs_j$, are formed.
The remaining data are simply assigned to their closest group. Thereby, we can guarantee the diversity of sensitive attributes within each group.
Once the input data of the sub-microdata Xid_j are partitioned into groups, of cardinality at least $k \times c_s$, the HM-PFSOM algorithm generates the anonymous sub-microdata X'_j, by replacing the data vectors by the centroid to which belong to (*line* 19).
The final anonymous microdata X' is considered as the union of the anonymous sub-microdata (*line* 20).

In the following, we propose to use an illustrative example to highlight the principle of the k-partitioning process, adopted by the HM-PFSOM algorithm.

Illustrative example

Let $X_{sub-micro}$, given in Table 4, be a sub-microdata of a given original microdata. Each input data $x_i \in X_{sub-micro}$ contains one confidential attribute, *i.e. Salary*.

Table 4: Example of sub-microdata $X_{sub-micro}$ generated from a given microdata

x_i	ZIP code	Age	**Salary**
x_1	1011	22	**500**
x_2	1007	22	**550**
x_3	1012	23	**600**
x_4	1009	25	**1600**
x_5	1010	28	**1500**
x_6	1011	29	**1800**
x_7	1013	31	**2900**
x_8	1010	32	**3200**
x_9	1008	32	**3600**
x_{10}	1010	29	**1650**
x_{11}	1009	26	**1550**
x_{12}	1011	27	**1700**
x_{13}	1008	33	**3800**

Before extracting the fixed size groups, the HM-PFSOM algorithm starts by applying a clustering process in order to study the distribution of the confidential attribute values.
We note that the confidential attribute of the input data are distributed into 3 clusters, which are respectively illustrated in Tables 5, 6 and 7. Indeed, the set of data $\{x_1,x_2,x_3\}$ are characterized by a salary varying between 500 and 600, *i.e.* a low salary values. While the salary attribute of the second set of data vectors $\{x_4,x_5,x_6,x_{10},x_{11},x_{12}\}$ ranges between 1500 and 1800, *i.e.* a middle salary values. Whereas the third set of data $\{x_7,x_8,x_9,x_{13}\}$ is characterized by high values of salary, varying between 2900 and 3800.

Table 5: Cluster 1: Data vectors with low salary

x_i	ZIP code	Age	**Salary**
x_1	1011	22	**500**
x_2	1007	22	**550**
x_3	1012	23	**600**

Table 6: Cluster 2: Data vectors with middle salary

x_i	ZIP code	Age	**Salary**
x_4	1009	25	**1600**
x_5	1010	28	**1500**
x_6	1011	29	**1800**
x_{10}	1010	29	**1650**
x_{11}	1009	26	**1550**
x_{12}	1011	27	**1700**

Table 7: Cluster 3: Data vectors with high salary

x_i	ZIP code	Age	**Salary**
x_7	1013	31	**2900**
x_8	1010	32	**3200**
x_9	1008	32	**3600**
x_{13}	1008	33	**3800**

To maintain the latter distribution of the confidential attribute, the HM-PFSOM algorithm builds the fixed size groups according to the obtained clusters, by requiring that each group should contain at least 3 *dissimilar* confidential attribute values. For example, the first fixed size groups G_1 gathers 3 data, namely x_1, x_8, x_{10} and x_{12}, belonging to different clusters. Indeed, the data vector x_1 has a *low* salary value, while the data vectors x_{10} and x_{12} are characterized, by a *middle* salary. Whereas, the data x_8 has a *high* salary. The same principle is applied to the other groups, *i.e.* G_2 and G_3. Thus, the final 3-partition should matches to the one given in Table 4.

To achieve such partition, we propose to fix at first the number of groups, denoted by k_g. Then, the data within each

Table 8: The optimal 3-partition of the sub-microdata $X_{sub-micro}$

	x_i	ZIP code	Age	**Salary**
G_1	x_1	1011	22	**500**
	x_{12}	1011	27	**1700**
	x_8	1010	32	**3200**
	x_{10}	1010	29	**1650**
G_2	x_2	1007	22	**550**
	x_4	1009	25	**1600**
	x_9	1008	32	**3600**
	x_{11}	1009	26	**1550**
G_3	x_3	1012	23	**600**
	x_5	1010	28	**1500**
	x_7	1013	31	**2900**
	x_6	1011	29	**1800**
	x_{13}	1008	33	**3800**

cluster are partitioned into the k_g groups. In our example, the number of groups is set equal to 3, and the data of the clusters *low*, *middle* and *high* salary are distributed into the 3 groups.

3.3 Fuzzy possibilistic clustering for hybrid microaggregation

The main challenge of the HM-PFSOM algorithm is to find the suitable set of the sub-microdata contained in the original microdata X, *i.e.* $X = \{Xid_1, Xid_2, \ldots, Xid_c\}$. Moreover, for each sub-microdata, it is important to find out its groups of confidential attributes and rightly assess their centres even in noisy surroundings. Clustering is a useful means to achieve such important target. Indeed, clustering aims to discover the unrevealed relationships between data, by splitting a dataset into disjoint clusters. Where each cluster gathers similar data, and dissimilar to those data in other clusters.
In this respect, we have proposed in a previous work a fuzzy possibilistic clustering algorithm, called Possibilistic Fuzzy Self Organising Map (PFSOM) (Abidi and Ben Yahia 2013), able to split a given dataset into c disjoint clusters, via a multi-level process. Where c corresponds to the optimal number of clusters contained in a dataset (Abidi et al 2012). Each level aims to group similar outputs resulting from the level below. Doing so, the first level is used to form an initial partition of the dataset X, by training the data into an initial clusters. The latter clusters are fine-tuned through a hierarchical levels. The role of each level is to build a partition of the outputs of the level below. Such a process produces a hierarchical structure composed of several partitions. To extract the best one, the PFSOM algorithm integrates, during the multi-level process, a validity index, called a *Partition Coefficient and Exponential Separation* (PCAES) index (Wu and Yang 2005). At each level, the algorithm assesses the quality of the partition, by evaluating the compactness and separation of its clusters. The optimal partition P corresponds to the one that have the best validity index. Subsequently, the number of clusters contained in the obtained partition P is equivalent to the optimal number of clusters c.
At each level, the clustering algorithm PFSOM relies on fuzzy possibilistic learning process, in order to decrease the influence of noisy data. In fact, real datasets are generally characterized by the presence of noisy data and outliers, which can directly influence the obtained data clusters. The PFSOM algorithm integrates both of the concept of typicality and membership values during the clustering process. In fact, to classify a data point, a cluster centroid has to be the closest one to the data point, and this what aims fuzzy clustering by using a probabilistic constraint, *i.e.* membership values (Bezdek et al 1984). In addition, for estimating the centroids, the possibilistic constraint, *i.e.* typicality values, is used for mitigating the undesirable effect of outliers (Krishnapuram and Keller 1993).
Generally speaking, to split a given dataset $X = \{x_1, x_2, \ldots, x_n\}$ into c clusters, the PFSOM algorithm starts by initializing the c cluster centres. Then, the prototypes of the latter are adjusted during a learning process. This means that, the estimation of the cluster centres is achieved through an iterative process. In each iteration, the prototype of each cluster center c_j is updated according to the membership and typicality values of all data to that cluster. Where, the membership value represents the degree to which a given data point x_i belongs to a cluster c_j. Such value is measured according to distances between x_i to all cluster centres. However, the typicality of a data point to a given cluster represents its resemblance to the other data points belonging to the same cluster, *i.e.* internal resemblance. The belonging of a data point x_i to a cluster c_j, depends on the distance from x_i to c_j relative to the distances of all data to that cluster (Pal et al 2005). The process of updating cluster centres as well as the membership and typicality values is repeated until the stability condition is fulfilled or the predefined number of iterations is achieved. Then, the clusters are obtained by assigning each data to its nearest center. The clustering process of the PFSOM algorithm is detailed in (Abidi and Ben Yahia 2013).

To sum up, the HM-PFSOM algorithm relies on two levels of clustering process. On each level, the PFSOM algorithm is used by adapting the distance measure. In fact, the first level consists in splitting the original microdata X into c disjoint sub-microdata according to their quasi-identifiers, *i.e.* $X = \{Xid_1, Xid_2, \ldots, Xid_c\}$. Thus, the distance used to compute typicality and membership values between a data vector x_i and a center c_j is defined as follows:

$$dist(x_i, c_j) = \sum_{l=1}^{Q} (x_i[qi_h] - c_j[qi_h])$$

Where Q refers to the number of quasi-identifiers, and $x[qi_l]$is the l^{th} quasi-identifier of a given data vector x. In the previous example given in Table 4, Q is equal to 2, while $x_i[qi_1]$ and $x_i[qi_2]$ correspond, respectively, to the *ZIP code* and *Age* of the data vector x_i. The same goes for the center c_j.
On the other hand, the second level of clustering process is used to study the distribution of the confidential attributes within each sub-microdata Xid_j. It is used to extract the c_s disjoint clusters of confidential attributes. So, the distance used by the PFSOM algorithm is defined as:

$$dist(x_i, c_j) = \sum_{p=1}^{S} (x_i[s_p] - c_j[s_p])$$

Where S refers to the number of confidential attributes, and $x[s_p]$ is the p^{th} sensitive attribute of a given data vector x. In the example given in Table 4, S is equal to 1, while $x_i[s_1]$ corresponds to the *salary* attribute of the i^{th} data vector.

4 Experimental results

This section aims to test the validity of the HM-PFSOM algorithm for generating an anonymous microdata. We evaluate the performance of our algorithm according to information loss and disclosure risk on well real microdata.
In the following we present firstly the manipulated microdata and the measures used to evaluate our algorithm. Then, we discuss the performance values of the HM-PFSOM algorithm.

4.1 Evaluation data

To evaluate the performance of our proposed HM-PFSOM algorithm with the main microaggregation methods, we consider the three real-world microdata, used as benchmarks in prior studies (Brand et al 2002), namely:

- *CENSUS*: This dataset was obtained on July 27, 2000 using the Data Extraction System of the U. S. Bureau of the Census. The *CENSUS* dataset contains 1080 records with 13 numeric attributes.
- *EIA*: This dataset was obtained from the U.S. Energy Information Authority. It consists of 4092 records with 15 attributes. Since the first two attributes are considered as direct identifiers of the records, we propose to de-identify the dataset by removing the latter attributes. Then, we have not taken into account the attribute $YEAR$, because the value of the latter attribute in the whole dataset is equal to 96. We also eliminated the categorical attribute $STATE$, given that our proposed algorithm is designed to handle continuous values. To sum up, we used in our experiment only the 11 remaining attributes.
- *TARRAGONA*: This real dataset comprises the figures from 834 companies in the area of Tarragona. This means that, the dataset contains 834 records with 13 numeric attributes.

Note that, in each dataset, the values of the attributes are well apart, *i.e.* range in different domains. This can distort the clustering results. Thus, we propose to normalize the manipulated datasets by Min-Max scaling technique. A value v of an attribute is normalized to the value v', by computing the following formula:

$$v' = \frac{v - min(v)}{max(v) - min(v)}$$

Where $min(v)$ and $max(v)$ refer, respectively, to the minimum and the maximum values of the attribute v.

4.2 Evaluation measures

Let X be an original dataset and X' its anonymous version. The quality of the microaggregation methods has been evaluated from the perspectives of information loss and disclosure risk, as follows:

- The information loss (IL) has been quantified by means of the well-known measure which was exposed in (Domingo-Ferrer and Torra 2004). The IL measure computes the mean variation between the original and the perturbed version of a record x_i, given by the following formula :

$$IL = \sum_{i=1}^{n} \left(\frac{1}{q} \times \sum_{j=1}^{q} \frac{|x_{ij} - x'_{ij}|}{\sqrt{2} S_j} \right) \quad (1)$$

  where S_j is the standard deviation of the j^{th} variable in the original data. Hence, the lower the information loss is, the higher the utility of the anonymous data.
- The disclosure risk (DR) is quantified by *Distance-Based Record Linkage* (DBRL). This method, introduced in (Pagliuca and Seri 1999), consists in computing distances between records in two datasets. The record linkage can be used to find out to what extent anonymous records could be re-identified. In general, for each record in the original dataset X, the distance to every record in the masked dataset X' is computed. Thereafter, the *nearest* and the *second nearest* records in X' are extracted. A record in the anonymous dataset X' is labeled as *linked*, when the nearest record in the original dataset X matches its corresponding original record. A record in the anonymous microdata X' is designed as linked to 2^{nd} *nearest*, if the second nearest record in X turns out to be the corresponding original record. In all other cases, the records are not linked.

4.3 Performance Analysis

The first part of the experimental evaluation consists in comparing the three main heuristics of microaggregation approach, namely:

- The univariate microaggregation by individual sorting.
- The univariate microaggregation based on single axis sorting criteria.
- The multivariate microaggregation based on diameter method, *i.e.* MDAV.

The aim of this comparative study is to support our choice of using the MDAV algorithm method in the fixed size partitioning process. Then, we discuss the performance of our proposed hybrid microaggregation algorithm.

4.3.1 Performance Comparaison of univariate and multivariate microaggregation methods

To assess the performance of the microaggregation methods, we used the *R-Package sdcMicro* (Templ et al 2015). The latter package includes the popular methods of generating protected microdata. In the remainder of this section, we choose to refer the univariate microaggregation heuristic based on individual sorting criteria by ONEDIMS, while the univariate microaggregation based on single axis sorting criteria, by PCA.

We evaluate the performance of the three main microaggregation heuristics on the real datasets cited above, by varying the value of the parameter k and the number of the quasi-identifiers. The information loss and disclosure risk measures of the different microaggregation heuristics are given in Figures 1, 2, 3, 4, 5 and 6.

By analyzing the results, we notice that the data utility and the disclosure risk are *inversely* proportional, regardless the microaggregation methods. We note that, for small values of k, the information loss is at minimum score, while the disclosure risk reaches its maximum scores. Otherwise, for high values of k the information loss values are increased and the anonymous dataset loses its utility, whereas the risk of records re-identification is minimized. Such a mismatch is confirmed by varying the privacy parameter k on EIA, Tarragona and Census microdata. This contradiction can be explained by the fact that for small values of k, the records of a given microdata are partitioned into a fixed size groups, *i.e.* of size k, gathering similar records. Thus, each record will be close to the centroid of the group to which it belongs to. Thus, replacing the quasi-identifiers of the original records by their centroid, will produce a modified microdata fairly close to the original one. However, this anonymous microdata can not be effective to hide the identity of original records, since an intruder may be able to link the anonymous records with their nearest original ones. On the other hand, by choosing a high value of k, the records of the original microdata will be partitioned into a reduced number of fixed size groups. Thereby, records with dissimilar quasi-identifiers will be *forced* to be gathered in a same group. By this way, the centroids of the fixed size groups will be miscalculated. Since the latter will be used to anonymize the microdata, the information loss will be significant, which can avoid the intruder to assume that the centroid assigned to a record is always the nearest one.

By examining the performance of the microaggregation heuristics, namely ONEDIMS, MDAV and PCA, we can note that the multivariate microaggregation, *i.e.* the MDAV method, is best suited than the univariate microaggregation, *i.e.* ONEDIMS and PCA methods. The performance is measured in terms of handling the conflicting principles, *i.e.* maintaining the data utility and avoiding the disclosure risk. Indeed, regardless the value of the parameter k, the ONEDIMS method results mostly lead to the lowest information loss, but the highest disclosure risk values. This can be explained by the fact that the univariate microaggregation by individual sorting criteria applies the anonymization process on each attribute. Thus, the anonymous multi-dimensional records can violate the k-anonymity property. On the other hand, the univariate microaggregation by PCA sorting criteria method, studies at first the underlying multidimensional *correlation structure* of the data vectors, to produce principal components. Then, the k-partition is formed by sorting the training data vectors according to their principal component. Thereafter, groups of successive k records are formed. However, ranking the records by such approach mainly rely on the correlation matrix. Such method can be very useful with data that are very highly correlated. Indeed, the higher the correlation the lower the information loss is. However, not all data are highly correlated.

To sum up, we can consider that the multivariate microaggregation is best suited than the univariate microaggregation, in terms of balancing between the two conflicting issues, namely data utility and privacy.

Several methods have been proposed to improve the performance of the MDAV method. However, these improvements have been addressed to decrease the information loss. As mentioned above, we think that the optimal k-anonymous microdata should not only maintain the information loss while fulfilling the k-anonymity property, but also it should avoid attribute disclosure. Thus, we have proposed the HM-PFSOM algorithm aiming to achieve the optimal k-anonymous microdata, as we have defined. We should remember that our proposed algorithm proceeds through an hybrid manner. It consists in splitting the microdata into disjoint sub-microdata, according to the similarity of the quasi-identifiers. Avoiding thus the risk of gathering records with dissimilar quasi-identifiers in a same group. Thereafter, the microaggregation process can be applied independently on each sub-microdata.

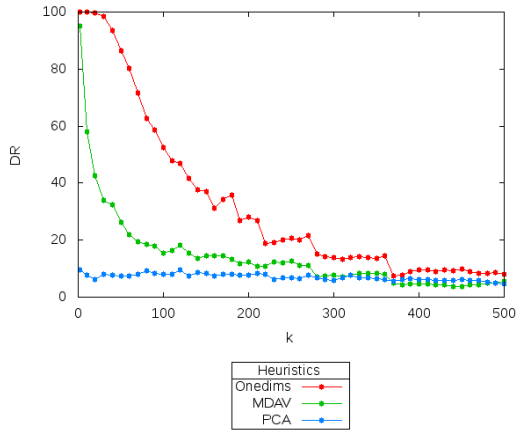

Fig. 1: Comparing the DR of the main microaggregartion heuristics on Census$_{|Qid|=5}$ microdata

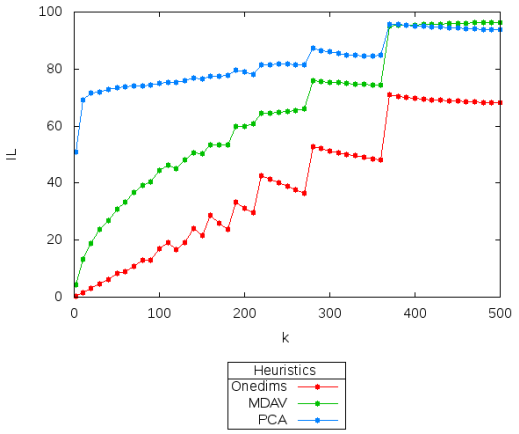

Fig. 2: Comparing the IL of the main microaggregartion heuristics on Census$_{|Qid|=5}$ microdata

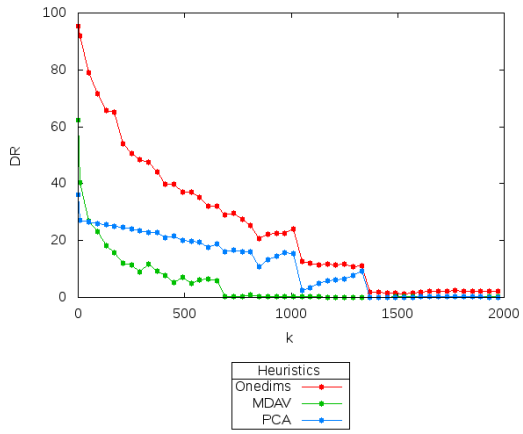

Fig. 3: Comparing the DR of the main microaggregartion heuristics on EIA$_{|Qid|=7}$ microdata

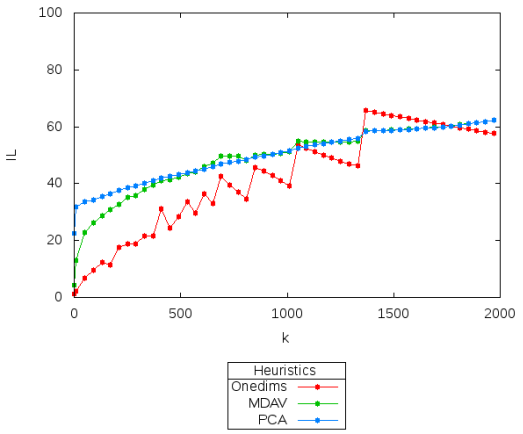

Fig. 4: Comparing the IL of the main microaggregartion heuristics on EIA$_{|Qid|=7}$ microdata

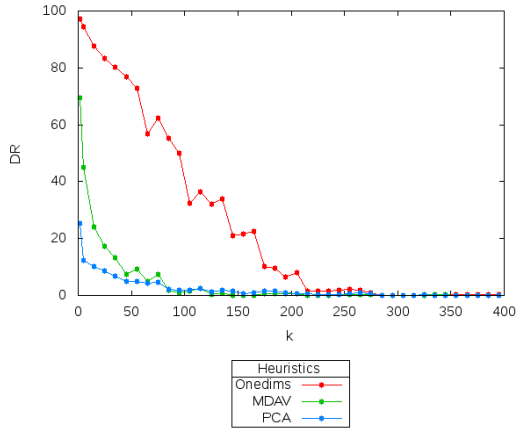

Fig. 5: Comparing the DR of the main microaggregartion heuristics on Tarragona$_{|Qid|=5}$ microdata

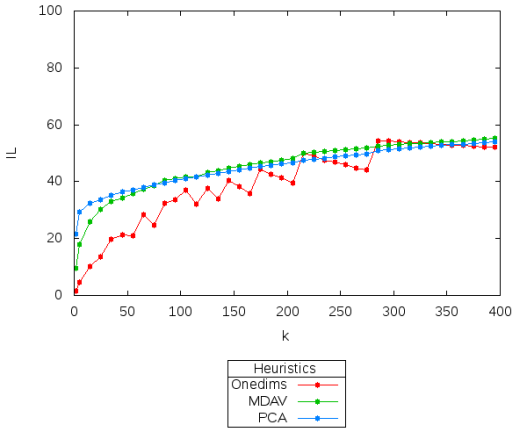

Fig. 6: Comparing the IL of the main microaggregartion heuristics on Tarragona$_{|Qid|=5}$ microdata

4.3.2 Performance analysis of hybrid microaggregation

In the following, we propose to compare the performance of the HM-PFSOM algorithm to those of the MDAV algorithm. The performances are evaluated in terms of *i)* homogeneity of sensitive attributes within the fixed size groups, by adapting the well-known *Sum of Squared Errors*; and *ii)* information loss (IL).
Let P be the k-partition of the original microdata X. The homogeneity of sensitive attributes within each group of the k-partition P is defined by :

$$SSE_i = \sum_{j=1}^{n_i} \sum_{p=1}^{s} dist(x_{jp}, cip), \forall i \in \{1, \ldots, g\} \quad (2)$$

Where SSE_i measures the distances between the sensitive attributes of the data belonging to a given group g_i to its centroid c_i. The latter centroid is determined by $\frac{\sum_{j=1}^{n_i} \sum_{p=1}^{s} x_{jp}}{n_i}$. Note that, the lower the value of SSE_i, the more the sensitive attributes within the i^{th} group are close to their centroid, which means that the sensitive attributes are similar. Since our aim is to ensure a diversity of sensitive attributes within the fixed size groups, thus high values of SSE_i indicate that the sensitive attributes of the i^{th} group are well *separated*.
By analyzing the experimental results exposed in Tables 10 and 12, we can notice that, regardless the predefined privacy parameter k, the HM-PFSOM algorithm is able to generate a fixed size partition, where the minimal size of its groups is proportional to the diversity of confidential attributes.
Suppose that the Census microdata is composed by a set of 5 quasi-identifiers and a set of 8 sensitive attributes. To anonymize the latter microdata, the HM-PFSOM algorithm starts by splitting the original microdata into 3 disjoint sub-microdata, according to the quasi-identifiers. We should mention that we have used the PFSOM to extract the latter sub-microdata.

Table 9 illustrates the 3 resulted sub-Census microdata, which contain respectively 382, 376 and 322 data records. On each sub-microdata, the HM-PFSOM algorithm studies in a second phase the distribution of the confidential attributes. For example, such as mentioned in Table 9, the data vectors of the first sub-microdata of Census, *i.e.* sub-Census$_1$, are in turn partitioned into 3 classes of sensitive attributes. While the second and the third sub-microdata contain respectively 4 and 2 classes of confidential attributes. These classes are then used to maintain the diversity within the obtained partition. For example, by setting the parameter k to the minimal value, *i.e.* $k = 2$, the obtained partition from sub-Census$_1$ is formed by 36 groups having a minimal size equal to 8. Indeed, each group contain at least 2 records from the three classes of confidential attributes. In fact, we note that the minimal homogeneity of sensitive attributes SSE_i within the obtained groups is equal 8.41. While the minimal SSE_i values obtained by the MDAV algorithm is equal to $0.9 \simeq 0$. This would mean that the k-partitioning process of the MDAV algorithm can group data within a same group having a *highly* similar sensitive attributes. These data are then exposed to the attribute attack. By increasing the value of the parameter k, the HM-PFSOM algorithm is able to maintain the diversity of sensitive attributes within the resulted partition. Indeed, by setting the parameter k to 25 the minimal SSE_i value obtained by HM-PFSOM algorithm is equal to 268.48. While that of the MDAV algorithm is equal to 35.56. However, as expected, ensuring a diversity of confidential attributes within the k-partition may affect the data utility. In fact, the performance of the information loss obtained by the MDAV algorithm are better than those of the HM-PFSOM algorithm. Indeed, on the three sub-microdata of Census microdata, by increasing the value of the privacy parameter, the information loss obtained by HM-PFSOM algorithm is increased from 22.75 to 40.31 for sub-Census$_1$; from 25.54 to 41.17 for sub-Census$_2$; and from 17.90 to 40.87 for sub-Census$_3$. Whereas, the information loss obtained by the MDAV algorithm increases from 8.84 to 26.03 for sub-Census$_1$, from 8.01 to 26.03 for sub-Census$_2$ and from 12.57 to 30.52 for sub-Census$_3$. The detailed observations are given in Table 10.
These low performances in terms of data utility, is also true for the sub-microdata tables of EIA microdata, which are illustrated in Table 12 (page 14). This can be explained by the fact that data with similar quasi-identifiers can be assigned to different groups, since they have a close sensitive attributes. By contrast, this can avoid the attribute disclosure risk.

5 Conclusion

In this paper, we have introduced a new algorithm for multivariate microaggregation HM-PFSOM, based on fuzzy possibilistic clustering to generate the optimal partition. The latter is used to generate an anonymous microdata. Hence, the HM-PFSOM algorithm covers three main goals, namely: *i)* Preventing the identity disclosure risk by requiring that the generated partition should fulfill the k-anonymity property; *ii)* Preventing the attribute disclosure by ensuring the diversity of confidential attributes within each fixed size group of the generated partition; *iii)* Maintaining the data utility of the anonymous microdata by increasing the homogeneity of the obtained partition.
The HM-PFSOM algorithm operates through an hybrid manner. Its main idea consists in splitting the original dataset into disjoint sub-datasets, in such a way that data within the same sub-dataset must be similar to some extend, also they should be dissimilar to those data in other sub-datasets. Such process enables to avoid the refinement phases used

Table 9: Distribution of the Census microdata

Microdata	$\|sub-microdata\|$	$\|Confidential\ classes\|$
Census	$\|sub-census_1\|$= 382	$\|C_1\|$= 145 $\|C_2\|$= 164 $\|C_3\|$= 73
	$\|sub-census_2\|$= 376	$\|C_1\|$= 50 $\|C_2\|$= 121 $\|C_3\|$=103 $\|C_4\|$=103
	$\|sub-census_3\|$= 322	$\|C_1\|$= 115 $\|C_2\|$= 207

Table 9 illustrates the 3 resulted sub-Census microdata, according to the 5 quasi-identifier attributes. Then, on each sub-Census microdata the distribution of confidential attributes is studied. For example, the data vectors of the first sub-Census$_1$ contains 3 classes of sensitive attributes. While the second and the third sub-microdata contain respectively 4 and 2 classes of sensitive attributes.

Table 10: Performance comparison on Census microdata

sub-microdata	k	minimal group size		# Groups		min $SSE_{j,1\leq j\leq cs}$		IL	
		HM-PFSOM	MDAV	HM-PFSOM	MDAV	HM-PFSOM	MDAV	HM-PFSOM	MDAV
sub-Census$_1$	1	4	1	73	382	4.46	0.00	22.75	0.00
	5	20	5	14	76	29.76	3.72	31.07	15.90
	10	40	10	7	38	67.92	10.90	32.96	20.75
	15	60	15	4	25	133.33	14.31	35.05	23.11
	20	82	20	3	19	149.11	23.63	36.84	24.82
	25	100	25	2	15	262.48	35.56	40.31	26.03
sub-Census$_2$	1	7	1	50	370	4.52	0.00	25.54	0.00
	5	35	5	10	75	27.26	2.66	31.24	13.98
	10	70	10	5	37	63.07	8.21	36.22	18.69
	15	105	15	3	25	104.27	12.44	37.05	20.64
	20	140	20	2	18	184.53	16.29	41.17	22.63
sub-Census$_3$	1	2	1	115	322	2.58	0.00	17.90	0.00
	5	10	5	23	64	13.07	6.30	25.89	16.30
	10	20	10	11	32	48.65	10.84	29.90	20.84
	20	40	20	5	16	151.60	30.60	33.37	24.74
	30	60	30	3	10	267.17	61.67	37.15	27.88
	40	80	40	2	8	445.00	72.94	40.87	30.52

The performances are evaluated in terms of information loss (IL) and diversity of sensitive attributes within the fixed size groups (SSE_i). The higher the value of SSE_i, the more the sensitive attributes within the i^{th} group are dissimilar. Thus, the attribute disclosure risk is low.

Table 11: Splitting the EIA microdata into disjoint sub-microdata according to the 7 quasi-identifiers.

Microdata	$\|sub-microdata\|$	$\|Confidential\ classes\|$
EIA	$\|sub-EIA_1\|$ = 464	$\|C_1\|$ = 238 $\|C_2\|$ = 78 $\|C_3\|$ = 148
	$\|sub-EIA_2\|$ = 1195	$\|C_1\|$ = 828 $\|C_2\|$ = 233 $\|C_3\|$ = 134
	$\|sub-EIA_3\|$ = 106	$\|C_1\|$ = 37 $\|C_2\|$ = 69
	$\|sub-EIA_4\|$ = 1191	$\|C_1\|$ = 273 $\|C_2\|$ = 918
	$\|sub-EIA_5\|$ = 1136	$\|C_1\|$ =259 $\|C_2\|$ = 877

Table 11 illustrates the 5 resulted sub-EIA microdata, according to the 7 quasi-identifiers of the data records. The third column shows the distribution of confidential attributes on each sub-EIA.

Table 12: Performance comparison on EIA microdata

sub-microdata	k	minimal group size		# Groups		min $SSE_{j,1\leq j\leq cs}$		IL	
		HM-PFSOM	MDAV	HM-PFSOM	MDAV	HM-PFSOM	MDAV	HM-PFSOM	MDAV
sub-EIA$_1$	1	5	1	78	464	4.89	0.00	21.21	0.00
	5	25	5	15	92	34.61	4.60	29.09	12.97
	10	50	10	7	46	88.69	8.74	33.44	17.51
	20	100	20	3	23	275.50	15.85	41.78	21.51
	30	150	30	2	15	457.89	42.64	46.45	23.84
sub-EIA$_2$	1	8	1	134	1195	3.15	0.00	50.47	0.00
	5	40	5	26	239	23.23	260	50.46	50.75
	10	80	10	13	11	62.14	5.73	50.46	50.46
	20	160	20	6	59	130.53	18.55	50.45	50.48
	30	240	30	4	39	262.95	27.02	50.45	50.46
	40	320	40	3	29	432.04	36.02	50.43	40.47
sub-EIA$_3$	1	2	1	37	106	8.54	0.00	26.07	0.00
	5	10	5	7	21	58.26	12.13	32.88	17.95
	10	20	10	3	10	134.27	29.50	40.44	25.49
	15	30	15	2	7	211.60	59.06	40.02	27.55
sub-EIA$_4$	1	4	1	273	1191	1.08	0.00	15.48	0.00
	5	20	5	54	238	7.33	0.10	18.52	5.18
	10	40	10	27	119	13.91	0.55	20.94	7.47
	30	120	30	9	39	72.60	2.68	25.25	11.08
	50	200	50	5	23	146.44	4.65	30.57	13.13
	100	400	100	2	11	474.85	10.74	46.75	17.22
sub-EIA$_5$	1	4	1	259	1136	1.09	0.00	14.69	0.00
	5	20	5	51	227	6.67	0.06	14.69	14.80
	10	40	10	25	113	15.68	0.61	14.64	14.76
	30	120	30	8	37	65.44	2.99	14.56	14.76
	50	200	50	5	22	137.34	4.28	14.50	14.72
	100	400	100	2	11	465.27	15.11	14.14	14.67

The performances are evaluated in terms of information loss (IL) and diversity of sensitive attributes within the fixed size groups (SSE_i). The higher the value of SSE_i is, the more the sensitive attributes within the i^{th} group are dissimilar.

to fine tune the k-anonymous partition, since the partitioning process is performed only on similar data. Indeed, HM-PFSOM applies the k-partitioning process independently on each sub-dataset. On privacy side, the HM-PFSOM approach proposes to study the distribution of confidential attributes within each sub-dataset. Then, according to the latter distribution, the privacy parameter k is determined, in such a way to preserve the diversity of confidential attributes within the anonymized microdata.

Our main interest in a future work lies in Privacy-Preserving Data Sharing in the Smart City. In fact, smart city is a vision proposed by many governments to integrate information and communication technology (ICT) solutions into the critical infrastructures of their cities and society with the goal of improving the quality of life of their citizens (Curry et al 2016). Smart city applications comprise a number of diverse areas, like smart card services for easy authentication and payment on the go, smart resource management of water or electricity, smart mobility applications that improve traffic efficiency and reduce CO_2 emissions (Novotny et al 2014). The effectiveness of these and other smart city applications heavily relies on data collection, interconnectivity, and pervasiveness. Smart city applications are increasingly relying on personally identifiable data. In fact, people's data are key elements in order to design effective and smart policies and services for citizens. Such type of data reflects the daily activities of the people living, working, and visiting the city, *e.g.* monitoring tourist foot traffic, or home energy usage, or homelessness (Zoonen 2016). The more connected a city the more it will generate a steady stream of data from and about its citizens. However, connected smart city devices raise concerns about individuals' privacy, autonomy, freedom of choice, and potential discrimination by institutions. Thus, privacy is a key concern in the facet of smart cities. Thereby, develop a new framework for exploring people's specific privacy concerns in smart cities is considered as an obvious prospect. The framework should hypothesize if and how smart city technologies and urban big data produce privacy concerns among the people in these cities, such as inhabitants, workers, visitors, and otherwise.

References

Abidi B, Ben Yahia S (2013) Multi-pfkcn : A fuzzy possibilistic clustering algorithm based on neural network. In: Proceedings of International Conference on Fuzzy Systems (FUZZ-IEEE 2013), Hyderabad, India, 7-10 July, 2013, IEEE, pp 1–8

Abidi B, Ben Yahia S, Bouzeghoub A (2012) A new algorithm for fuzzy clustering able to find the optimal number of clusters. In: Proceedings of 24th International Conference on Tools with Artificial Intelligence, ICTAI 2012, Athens, Greece, November 7-9, 2012, IEEE, pp 806–813

Aggarwal CC, Yu PS (2008) An introduction to privacy-preserving data mining. In: Privacy-Preserving Data Mining - Models and Algorithms, Advances in Database Systems, vol 34, Springer, pp 1–9

Agrawal R, Srikant R (2000) Privacy-preserving data mining. ACM SIGMOD Record 29(2):439–450

Bennardo A, Pagano M, Piccolo S (2015) Multiple bank lending, creditor rights, and information sharing. Review of Finance

Bezdek JC, Ehrlich R, Full W (1984) FCM: The fuzzy c-means clustering algorithm. Computers and Geosciences 10(2-3):191–203

Brand R, Domingo-Ferrer J, Mateo-Sanz JM (2002) Reference data sets to test and compare sdc methods for protection of numerical microdata. Tech. rep., Computational Aspects of Statistical Confidentiality

Chang C, Li Y, Huang W (2007) Tfrp: An efficient microaggregation algorithm for statistical disclosure control. Journal of Systems and Software 80(11):1866–1878

Chen K, Liu L (2008) A survey of multiplicative perturbation for privacy-preserving data mining. In: Privacy-Preserving Data Mining - Models and Algorithms, Advances in Database Systems, vol 34, Springer, pp 157–181

Chittaranjan C, Blom J, Gatica-Perez D (2013) Mining large-scale smartphone data for personality studies. Personal Ubiquitous Computing 17(3):433–450

Chui M, Farrell D, Jackson K (2014) How government can promote open data. Tech. rep., McKinsey Global Institute

Ciriani V, di Vimercati SDC, Foresti S, Samarati P (2007) Microdata protection. In: Secure Data Management in Decentralized Systems, Advances in Information Security, vol 33, Springer, pp 291–321

Curry E, Dustdar S, Sheng QZ, Sheth A (2016) Smart cities – enabling services and applications. Journal of Internet Services and Applications 7(1)

Domigo-Ferrer J, Solanas A, Martínez-Ballesté A (2006) Privacy in statistical databases: k-anonymity through microaggregation. In: Proceedings oh the IEEE International Conference on Granular Computing, GrC 2006, Atlanta, Georgia, USA, May 10-12, 2006, pp 774–777

Domingo-Ferrer J (2008) A survey of inference control methods for privacy-preserving data mining. In: Privacy-Preserving Data Mining, vol 34, Springer, pp 53–80

Domingo-Ferrer J, Úrsula González-Nicolás (2010) Hybrid microdata using microaggregation. Information Sciences 180(15):2834–2844

Domingo-Ferrer J, Mateo-Sanz JM (2002) Practical data-oriented microaggregation for statistical disclosure control. IEEE Transactions on Knowledge and Data Engineering (TKDE) 14(1):189–201

Domingo-Ferrer J, Torra V (2004) Disclosure risk assessment in statistical data protection. J Comput Appl Math 164-165(1):285–293

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

Du W, Zhan Z (2003) Using randomized response techniques for privacy-preserving data mining. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, KDD '03, pp 505–510

Evfimievski A, Srikant R, Agrawal R, Gehrke J (2002) Privacy preserving mining of association rules. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, KDD '02, pp 217–228

Garfinkel SL (2015) De-identification of personal information. Tech. rep., National Institute of Standards and Technologie

Hundepool A, Domingo-Ferrer J, Franconi L, Giessing S, Nordholt ES, Spicer K, de Wolf P (2012) Statistical Disclosure Control. Wiley

Johnson M, Egelman S, Bellovin SM (2012) Facebook and privacy: It's complicated. In: Proceedings of the Eighth Symposium on Usable Privacy and Security, ACM, SOUPS '12, pp 9:1–9:15

Kargupta H, Datta S, Wang Q, Sivakumar K (2005) Random-data perturbation techniques and privacy-preserving data mining. Knowledge and Information Systems 7(4):387–414

Krishnapuram R, Keller J (1993) A possibilistic approach to clustering. IEEE Transactions on Fuzzy Systems 1(2):98–110

Lin J, Wen T, Hsieh J, Chang P (2010) Density-based microaggregation for statistical disclosure control. Expert Systems With Applications 37(4):3256–3263

Liu K, Giannella C, Kargupta H (2008) A Survey of Attack Techniques on Privacy-Preserving Data Perturbation Methods, Springer US, pp 359–381

Machanavajjhala A, Kifer D, Gehrke J, Venkitasubramaniam M (2007) *L*-diversity: Privacy beyond *k*-anonymity. ACM Transactions on Knowledge Discovery from Data (TKDD) 1(1):3

Martínez-Ballesté A, Solanas A, Domingo-Ferrer J, Mateo-Sanz JM (2007) A genetic approach to multivariate microaggregation for database privacy. In: Proceedings of the 23rd International Conference on Data Engineering Workshops, ICDE 2007, 15-20 April 2007, Istanbul, Turkey, pp 180–185

Matwin S (2013) Privacy-Preserving Data Mining Techniques: Survey and Challenges, Springer Berlin Heidelberg, pp 209–221

Mivule K (2013) Utilizing noise addition for data privacy, an overview. Computing Research Repository (CoRR)

Nin J, Torra V (2009) Analysis of the univariate microaggregation disclosure risk. New Generation Comput 27(3):197–214

Nin J, Herranz J, Torra V (2008) On the disclosure risk of multivariate microaggregation. Data & Knowledge Engineering (DKE) 67(3):399–412

Novotny R, Kuchta R, Kadlec J (2014) Smart city concept, applications and services. Journal of Telecommunications System & Management 3(2)

Oganian A, Domingo-Ferrer J (2001) On the complexity of optimal microaggregation for statistical disclosure control. Statistical Journal of the United Nations Economic Comission for Europe 18:345–354

Ohm P (2010) Broken promises of privacy: Responding to the surprising failure of anonymization. UCLA Law Review 57(6):1701–1777

Pagliuca D, Seri G (1999) Some results of individual ranking method on the system of enterprise accounts annual survey. Tech. rep., Esprit SDC Project

Pal NR, Pal K, Keller JM, Bezdek JC (2005) A possibilistic fuzzy c-means clustering algorithm. IEEE Transactions on Fuzzy Systems 13(4):517–530

Peersman G (2014) Overview: Data collection and analysis methods in impact evaluation. Methodological briefs - impact evaluation no. 10, UNICEF Office of Research

Rider AK, Chawla NV (2013) An ensemble topic model for sharing healthcare data and predicting disease risk. In: Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics, ACM, BCB'13, pp 333–340

Solanas A, González-Nicolás Ú, Martínez-Ballesté A (2012) Mixing genetic algorithms and V-MDAV to protect microdata. In: Computational Intelligence for Privacy and Security, pp 115–133

Solon O (2018) Facebook says cambridge analytica may have gained 37m more users' data. URL

Sweeney L (2002) K-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10(5):557–570

Templ M (2008) Statistical disclosure control for microdata using the r-package sdcmicro. transactions on Data Privacy 1(2):67–85

Templ M, Kowarik A, Meindl B (2015) Statistical disclosure control for micro-data using the r package sdcmicro. Journal of Statistical Software 67(4)

Teplitzky S (2014) Open data, [open] access: Linking data sharing and article sharing in the earth sciences. Journal of Librarianship and Scholarly Communication

Wu K, Yang M (2005) A cluster validity index for fuzzy clustering. Pattern Recognition Letters 26(9):1275–1291

Zoonen L (2016) Privacy concerns in smart cities. Government Information Quarterly 33