An overview of the use of clustering for data privacy

Vicenç Torra, Guillermo Navarro-Arribas, and Klara Stokes

Abstract In this chapter we review some of our results related to the use of clustering in the area of data privacy. The paper gives a brief overview of data privacy and, more specifically, on data driven methods for data privacy and discusses where clustering can be applied in this setting. We discuss the role of clustering in the definition of masking methods, and on the calculation of information loss and data utility.

Key words: data privacy, clustering, fuzzy clustering, information loss, microaggregation.

1 Introduction

Data privacy has emerged as an important area of research in the last years due to the increasing amount of information available that contains sensitive data from people and companies. Privacy preserving data mining (PPDM) and statistical disclosure control (SDC) are the two areas which study methods and tools to ensure that disclosure does not take place.

Methods for data privacy can be classified into different categories, and there exist different approaches for this classification. One of them is according to the information on the type of calculation that the receptor of the data (a third party) will apply to the data. Under this categorization we can distinguish between computation-driven, data-driven and result-driven approaches.

V. Torra
University of Skövde, Skövde, e-mail: vtorra@his.se

G. Navarro-Arribas
Universitat Autònoma de Barcelona, e-mail: guillermo.navarro@uab.cat

K. Stokes
University of Skövde, Skövde, e-mail: klara.stokes@his.se

Computation-driven methods are defined taking into account which is the analysis to be applied to the data. Data-driven methods are defined when the detailed analysis is unknown. Result-driven focuses on the sensitivity of the outcomes of the analysis. In this paper we focus on data-driven methods.

Data-driven methods for databases typically consist on modifying a database reducing its quality so that sensitive information is not disclosed. The modification should be in a way that the analyses on the modified data are similar to the analysis on the original data. Formally, if X is the original information, we have a method ρ such that when applied to X leads to a file X' that is quite similar to X but with less disclosure risk. Methods ρ of this characteristics are known as masking methods. Figure 1 shows the typical scenario of data-driven protection methods.

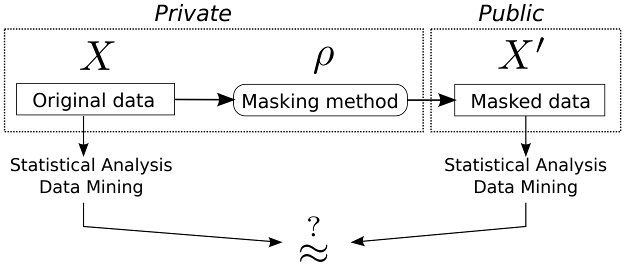

Fig. 1: Common scenario for data-driven protection methods.

Three main topics of research are of interest for data-driven methods. They are (i) masking methods (this is to answer which are the effective methods for data protection), (ii) disclosure risk measures (how we evaluate that the modified database X' is appropriate to ensure confidentiality), (iii) information loss measures (how we evaluate that the perturbation is not too high to make analysis useless).

Masking methods ρ can be classified into three main categories. The first one corresponds to methods that modify the original data introducing some kind of error. That is, $X' = X + \varepsilon$. In these methods, records in X' will contain some incorrect information. For example, salaries of individuals are lower or larger than the real ones. This category corresponds to perturbative methods and includes noise addition, microaggregation, and rank swapping. The second class which correspond to non-perturbative methods is defined by methods that do not produce erroneous data but change the level of detail. For example, salaries can be replaced by intervals, and cities by counties or regions. No correct value is replaced by an incorrect one. Generalization and suppression are the typical examples of non-perturbative methods. The third category corresponds to synthetic data generators. That is, the original data is replaced by synthetic data which follow a certain model that approximates the original data.

In this chapter we focus on data-driven approaches and we review masking methods based on clustering and information loss measures based on clustering. Clustering has an important role in both the definition of masking methods and the measure of information loss. More specifically, microaggregation is a well known perturbative masking method based on clustering that we will discuss it in Section 2. In addition, clustering has been used extensively to evaluate the quality of protected data. We will discuss clustering to measure information loss in Section 3.

In addition to the topics explained in this chapter, clustering has also been studied in computation-driven approaches. That is, when we know that the third party will cluster the data set. In this framework, the typical scenario is that a few data owners (e.g., companies) want to apply a clustering algorithm to their data but without sharing their records. To do so, a cryptographic protocol is established so that the resulting set of clusters are computed without revealing the original records to the other data owners. Different algorithms exists. Some algorithms presume that the data is vertically partitioned and others that it is horizontally partitioned. That is, data owners have information on different variables from the same people or the same variables from different people. See e.g. [57] for details.

2 Clustering to define masking methods

As explained in the previous section, masking methods are functions that introduce some distortion to the data in order to protect sensitive information.

2.1 Clustering in microaggregation

Microaggregation [10, 13, 17] is one of the methods for data protection. It has been proven [14, 15, 16] to be effective for data protection as it permits us to obtain a good trade off between information loss and disclosure risk.

Given a data set, microaggregation consists on building small clusters and then replacing the data by the cluster representatives. Privacy is achieved because we require that each cluster contains at least k records where k is a parameter of the method. The larger the k, the more privacy we have. Nevertheless, a large k also implies a large information loss. Because of that, microaggregation algorithms try to find a good tradeoff between privacy and information loss by means of an approapriate value for k. Formally, this method is defined by the following optimization problem. In this definition we have that $x \in X$ are the records, p_i is the centroid of the ith cluster, and $\chi_i(x) = 1$ represents that record x is assigned to the ith cluster. The application of the algorithms requires that we have a distance function between records and cluster centers, and the value k which is the minimum number of records in a cluster. We denote as $d(x_j, p_i)$ the distance between record x_j and centroid p_i. Equation (1) shows the formalization microaggregation as an optimization problem, minimizing the distance between records of a given cluster with their centroid, subject to the constraint imposed by the k parameter regarding the size of the clusters.

$$\begin{aligned}
\text{Minimize} \quad & \sum_{i=1}^{c}\sum_{j=1}^{n} \chi_i(x_j)(d(x_j,p_i))^2 \qquad (1)\\
\text{Subject to} \quad & \sum_{i=1}^{c} \chi_i(x_j) = 1 \text{ for all } j = 1,\ldots,n\\
& 2k \geq \sum_{j=1}^{n} \chi_i(x_j) \geq k \;\text{ for all } i = 1,\ldots,c\\
& \chi_i(x_j) \in \{0,1\}
\end{aligned}$$

Id	Age	Income
885	24	21000.00
795	31	19500.00
295	32	22000.00
058	57	43480.00
732	49	39220.00
925	43	32285.00
465	39	40500.00
321	20	20000.00
223	51	43050.00

(a) Original microdata

Id	Age	Income
–	25.00	20166.67
–	25.00	20166.67
–	38.00	31595.00
–	52.33	41916.67
–	52.33	41916.67
–	38.00	31595.00
–	38.00	31595.00
–	25.00	20166.67
–	52.33	41916.67

(b) Masked microdata with microaggregation for $k = 3$.

Table 1: Example of microaggregation.

Table 1 shows a simple example of microaggregation applied to numerical continuous attributes. The resulting masked table, composed of 3 clusters, is 3-anonymous. As it is a common practice in data privacy, identifiers are removed.

Microaggregation algorithms have been proven to be NP-hard problems [41] except for the case of a single variable (univariate microaggregation). A polynomial algorithm exists for this problem [22] and for some variants (e.g., univariate microaggregation with data supression [27]).

Microaggregation was originally defined for data represented as records on a set of numerical variables [10], and later extended to categorical variables [53]. Currently, there are extensions and variations for other types of structures as search and access logs, time series, documents, and graphs.

All heuristic algorithms for microaggregation follow the same pattern. First, data is clustered so that records are assigned to clusters and each cluster has at least k records. This is the clustering step and for this purpose a distance is needed in the space of the original data. Then, a cluster representative is selected from the cluster. For this purpose aggregation operators [55] are typically used. When data is numerical it is usual to use the mean while other operators as the median and the mode are used for non numerical data. Finally, the original data is replaced by the cluster representative.

Documents, or, in general, categorical information that can be interpreted semantically permits us to consider semantic versions of microaggregation. Note that as clustering algorithms are based on distances, and that it is usual to consider different types of distances. When data is categorical, we can use syntactic distances but also distances based on the semantics between terms. Semantic distances based on Wordnet [19] and on the Open Directory Project [11] have been considered in microaggregation. This is discussed in more detail below.

Microaggregation is related to *k*-anonymity [43, 52] as the application of microaggregation to a data set considering all the variables at the same time with a certain given *k* will satisfy *k*-anonymity.

2.2 Clustering for graphs: microaggregation and k-anonymity

The underlying structure of a social network is a graph, where nodes represent the individuals in the network and the edges their connections. In addition to the connectivity, both the nodes and the edges can contain additional information about the individuals and their relationships.

Masking methods for data protection for graphs can be classified using the same classes that exist for data files. There are perturbative methods that e.g. modify the graph adding and removing edges and vertices. In addition, non-perturbative methods reduce the graph into a kind of supernodes which in some sense generalize the connections between the original nodes.

The similarities and differences between *k*-anonymity for graphs and *k*-anonymity for standard files are discussed in [51]. This discussion follows the arguments in [59] to state the difficulty of working with graphs. However, in general, as [51] points out, every type of data has its own peculiarities.

Different masking methods for graphs consider different types of attacks. There are methods [32] that presume that the information available to an intruder is the number of connections of a node (e.g., the number of friends in a social network). This corresponds to the degree of the nodes. Others assume that the intruder knows a subgraph (some relationships between nodes) or, in general, a certain type of query on the graph [23]. See also, [59, 20] for other types of definitions. [51] reviews reidentification and *k*-anonymity definitions for graphs.

Given a graph $G = (V,E)$ where V are the nodes and E the edges of a graph ($E \subseteq V \times V$), [51] defines *k*-anonymity for graphs in terms of the neighbors of a node. The set of neighbors of a node $v \in V$ is defined as

$$N(v) := \{u \in V : (u,v) \in E\}.$$

Then, *k*-anonymity for graphs is defined as follows, see [51].

Definition 1. Let $G = (V,E)$ be a graph; then, we say that G is *k*-anonymous if for any vertex v_1 in V, there are at least k distinct vertices $\{v_i\}_{i=1}^{k}$ in V, such that $N(v_i) = N(v_1)$ for all $i \in \{1,\ldots,k\}$.

This definition ca be applied when the intruder knows (some of) the neighbors of a node.

0	0	0	1	1	1	0	0	0	1	1	1	1	1	1
0	0	0	1	1	1	0	0	0	1	1	1	1	1	1
0	0	0	1	1	1	0	0	0	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0	1	1	1	0	0	0
1	1	1	0	0	0	0	0	0	1	1	1	0	0	0
1	1	1	0	0	0	0	0	0	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
1	1	1	1	1	1	0	0	0	0	0	0	1	1	1
1	1	1	1	1	1	0	0	0	0	0	0	1	1	1
1	1	1	1	1	1	0	0	0	0	0	0	1	1	1
1	1	1	0	0	0	1	1	1	1	1	1	0	0	0
1	1	1	0	0	0	1	1	1	1	1	1	0	0	0
1	1	1	0	0	0	1	1	1	1	1	1	0	0	0

Table 2: Adjacency table of a simple 3-anonymous graph.

Clustering algorithms have been used in masking methods for graphs. Standard clustering algorithms for numerical data can be used for ensuring *k*-degree anonymity (i.e., that a given degree sequence is *k*-anonymous). In contrast, specific algorithms for graphs have been used to cluster the nodes of a graph to build *k*-anonymous graphs. In this case the goal is to build a graph that is *k*-anonymous in the sense of Definition 1. Figure 2 gives a small example of a 3-anonymous graph whose adjacency matrix is given in Table 2. It can be seen that for each node (each row) there are other 2 which have the same neighbors. Most algorithms for the clustering of social networks are centralized. That is, it is assumed that they are applied by the data owner who has all the data. [47] presents a distributed approach of message passing type.

[50] discusses the difference between two different approaches for graph partitioning: direct and indirect partitioning.

- Direct partitioning. This consists on partitioning the original matrix that represents the graph. This implies that clusters are built gathering together sets of nodes that have a good connectivity among themselves. This approach does not permit us to distinguish between the different roles of the vertices in well connected regions.
- Indirect partitioning. In this case, the partitioning algorithm is not applied to the graph (the original matrix) but to a similarity matrix computed from the graph. That is, given a similarity function S we build a matrix $M_S : V \times V \to [0,1]$ defining $M_s(V_1, V_2) = S(V_1, V_2)$. Then, we partition M_S.

It is clear that while the direct partitioning gathers in the same class connected nodes, the indirect partitioning gathers in the same class nodes that are similar.

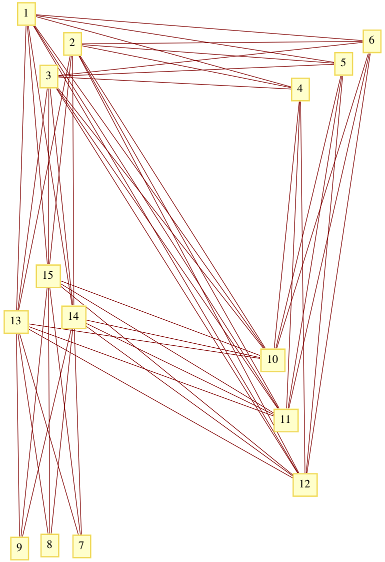

Fig. 2: Example of a simple 3-anonymous graph.

2.3 Attacks on microaggregation

When a data set is protected using microaggregation and taking all the variables at the same time in the clustering, the microaggregated file satisfies k-anonymity. In this case, attacks for k-anonymity are of relevance. They are [30, 9] the homogeneity and the background attacks.

Nevertheless, microaggregation is also applied to subsets of variables. This is used to decrease information loss at the cost of some disclosure risk. This implies that k-anonymity is not guaranteed. Table 3 illustrates the application of optimal microaggregation to each variable independently.

In this case, intruder can use the values of the masked file as well as their own information to attack the data set. In this example we can see that there are two unique records in the masked data set. The real k-anonymity of the protected file is one. Here, we use real k-anonymity as first defined in [39]. So, effective re-identification attacks can be done to this file. In general, as microaggregation modifies the original data, re-identification does not need to be straightforward, and we may have some records for which the nearest masked one is the correct link but others for which is not true. Nevertheless, intersection attacks are possible combining the information the intruder has for each each of the variables.

This type of intersection attack was first considered in [54] and later in [38, 40]. These latter works show empirically that some microaggregation methods fail to protect the data file.

Id	Age	Income	Id	Age	Income
885	24	21000.00	–	25.00	20166.67
795	31	19500.00	–	25.00	20166.67
295	32	22000.00	–	38.00	31168.33
058	57	43480.00	–	52.33	42343.33
732	49	39220.00	–	52.33	31168.33
925	43	32285.00	–	38.00	31168.33
465	39	40500.00	–	38.00	42343.33
321	20	20000.00	–	25.00	20166.67
223	51	43050.00	–	52.33	42343.33

(a) Original microdata

(b) Masked microdata with microaggregation for $k = 3$ applying optimal microaggregation to each variable.

Table 3: Example of microaggregation.

This type of attacks are related to the idea of transparency in data privacy. We have transparency when a release of a file goes with all the information on how the data is produced. This includes information on the masking method applied as well as its parameters.

2.4 Fuzzy clustering for microaggregation

Most methods for microaggregation are based on crisp clustering methods. In order to avoid some of the attacks mentioned in the previous section fuzzy clustering [5, 35, 36] was introduced in [54] as the clustering algorithm for microaggregation. Recall that the idea behind fuzzy clustering is that records can belong to more than one cluster.

In this approach, the assignment of records to clusters is not deterministic. Instead, it is done probabilistically according to a probability distribution. This probability can be proportional to the membership degrees of records to clusters or just uniformly distributed for clusters with membership above a certain threshold.

The goal of using fuzzy clustering and the random selection is to avoid intersection attacks when the different variables are considered. In addition, an intruder cannot be sure that the nearest masked record will be the one that correctly matches to the one in its own database.

2.5 Clustering for masking data streams

The application of microaggregation directly to mask data streams is usually not recommended. If microaggregation is applied using a sliding (or tumbler) window, thus applying microaggregation by parts of the stream, the result might be very poor in terms of information loss. Moreover, if not done carefully, e.g. by allowing recomputation of centroids after publication, it can be vulnerable to inference attacks through intersection [49, 6].

Stream clustering methods for data privacy differ from common stream clustering techniques in several points. Most notably, the main objective of the masking method is to produce the masked output, not the partition or structure of the clustering. This makes methods based on coresets, or in general techniques that require adjusting the clusters parameters as the stream is processed, not suitable for masking. Several techniques have been developed with this constraints in mind [6, 8, 56, 42]. These are streaming clustering methods that can implement k-anonymity in data streams, while avoiding disclosure from intersection of clusters.

Some ideas behind stream masking based on clustering have been extended to support fully dynamic data. Allowing the deletion of already masked data imposes an important threat. For instance if a a cluster is left with less than k elements these element need to be protected. This protection is difficult since has to prevent inferences. There are some works providing dynamic clustering and microaggreation as a masking method [58, 37], which still present some important drawbacks for example regarding information loss. The protection of dynamic data for publication is still an open research field.

2.6 Masking very large data sets

Although the stream masking methods discussed in the previous section can be used to mask high volumes of data, there are specific approaches to deal with this problem without the restrictions imposed by streaming data. These proposals improve generic microaggregation algorithms which need to access the whole data set during the masking process repetitively.

Some efficiency improvements can be achieved by projecting the data into one dimension and performing an optimal microaggregation [21], or by using specific data structures [24, 29]. Other approaches define an initial partition of the data in order to apply microaggregation in each part separately [44, 45].

Very efficient microaggregation can also be achieved by defining the clustering using k-nearest neighbors searches [46]. [28] is another recent approach based on local search.

Note that common microaggregation algorithms such as MDAV [13] present a complexity of $O(n^2)$ (where n is the number of records), which is unaffordable for very large data sets. The previously cited works reduce this complexity at least for some specific cases.

2.7 Masking through semantic clustering

As previously mentioned microaggregation can be defined for categorical data exploiting their semantics. This is very convenient for data privacy since precisely the semantics of the data is the important part to be preserved when masking data. These methods usually achieve a better compromise between privacy and information loss than syntactic approaches.

Microaggregation can be defined in terms of a semantic distance and a semantic aggregation operator to compute the clusters representatives. An example is to use an ontology such as Wordnet to define the distance and to aggregate words or synsets by means of generalization [1, 31, 34]. Note that here generalization is from the point of view of semantics (e.g., dog and cat are generalized into pets) and the dictionary can be used for this purpose.

This approach can be extended to deal with document vectors (algebraic representation of documents widely used in information retrieval and text mining) providing anonymous document vector spaces using microaggregation by clustering the vectors [37]. Although it is not a semantic microaggreation strictly speaking, spherical microaggregation has also been introduced to deal with document vectors [2].

Semantic microaggregation has also been applied to the anonymization of query logs from a search engine [18, 4]. In this case the Open Directory Project is used to semantically categorize queries (based on their actual results), and semantic distances are computed over those categories for clustering user queries. The semantic anonymization of set valued data has also been treated in [3].

2.8 Clustering in other masking methods

Clustering has been also used to define other masking methods. It is worth to mention its application to build data models that are accurate on subdomains. For example in [12, 48] data is clustered in a first step and then masked data is generated whithin the clusters. In this way, properties of the data at the cluster level can be preserved. For example, as microaggregation preserves mean, means will be preserved in the clusters if microaggregation is used for masking data in the second step. Similarly, if we use rank swapping in the second step, as rank swapping preserves frequencies, frequencies will be preserved in the clusters. [12] follows a different approach, it uses microaggregation in the first step, and then a synthetic data generator for the second step. In this way, the first step ensures a certain privacy level through the selection of the value of k. When $k = 1$ the original data is retrieved. So, there is no information loss and the risk is maximal. When $k = |X|$, protection is maximal and X is replaced by data according to the synthetic data generator for the full data set X. Note that this is different to what we obtain with microaggregation directly applied to the file. In such case, for $k = |X|$ we have that all records are replaced by the mean of the whole file X (i.e., all masked records are equal to the mean of X).

3 Clustering to measure information loss

Information loss depends on the data use. That is, on the analysis or function that the user intends to apply to the data. Naturally, the results of an analysis are different when data sets are different. Therefore, the analysis on the protected data set and on the original data set are different. The more perturbation the masking method applies to the data, the larger the difference between the original and the protected data set, and the larger the information loss.

Information loss measures can be formalized as follows. If f is the analysis to be applied to the data, X the original data set and X' the protected one obtained as the application of a masking method ρ to X (i.e., $X' = \rho(X)$), the information loss for a particular use f is the measure

$$IL(X,X') = divergence(f(X), f(X'))$$

where *divergence* is a function that quantifies the difference between $f(X)$ and $f(X')$. Naturally, *divergence(Y,Y)=0.*

Different types of analysis f have been considered in the literature. Clustering is one of them. Both crisp and fuzzy clustering have been considered, and the corresponding information loss has been measured. For example, information loss for *k*-means and fuzzy *c*-means have been measured for a few masking methods as e.g. microaggregation.

In the case of crisp clustering, divergence is a function that needs to consider the results of the clustering algorithm on the original file (i.e., $f(X)$) and on the protected file (i.e., $f(X')$). In this case, these results $f(X)$ and $f(X')$ are two partitions of the elements in X. Therefore any distance or similarity measure on pairs of partitions can be used to define the divergence. For example, we can use the Rand or the Jaccard index for this purpose. When $f(X)$ is a partition $\Pi = \{\pi_1, \ldots, \pi_n\}$ and $f(X')$ is the partition $\Pi' = \{\pi'_1, \ldots, \pi'_n\}$, the Rand and Jaccard indices are defined by:

Rand index:

$$RI(\Pi, \Pi') = (r+u)/(r+s+t+u)$$

Jaccard Index:

$$JI(\Pi, \Pi') = r/(r+s+t)$$

Adjusted Rand Index: This is a correction of the Rand index so that the expectation of the index for partitions with equal number of objects is 0. This adjustment was done assuming generalized hypergeometric distribution as the model of randomness. That is,

$$ARI(\Pi,\Pi') = \frac{r - exp}{max - exp}$$

where $exp = (np(\Pi)np(\Pi'))/(n(n-1)/2)$ and where $max = 0.5(np(\Pi)+np(\Pi'))$.

In these indices, r, s, t, u, and $np(\Pi)$ are defined as follows:

- r is the number of pairs (a,b) where a and b are in the same cluster in Π and in Π';
- s is the number of pairs where a and b are in the same cluster in Π but not in Π';
- t is the number of pairs where a and b are in the same cluster in Π' but not in Π;
- u is the number of pairs where a and b are in different clusters in both partitions.
- $np(\Pi)$ is the number of pairs within clusters in the partition π.

The Rand index is 1 when the two partitions are equal and 0 when the difference is maximal. Therefore, we can define

$$IL_{RI}(X,X') = divergence_{RI}(f(X), f(X')) = 1 - RI(f(X), f(X')).$$

The Adjusted Rand Index has the same behavior as the Rand Index, but has an expected value of zero and can take negative values. Therefore, we can also use in this case:

$$IL_{ARI}(X,X') = divergence_{ARI}(f(X), f(X')) = 1 - ARI(f(X), f(X')).$$

The Jaccard index can be used in the same way.

In the case of fuzzy clustering, $f(X)$ and $f(X')$ will be fuzzy partitions. Therefore, we can use here distances and generalization of these indices for fuzzy partitions. See [7, 25] for details.

These indices have been used e.g. in [26] to compare the results of masking method with respect to the use of clustering.

Examples the use of clustering as an information loss measure in the particular case of semantic-based masking methods can be found in [4, 33].

4 Conclusion

In this chapter we have discussed the application of clustering in data privacy. We have seen that clustering is applied in the definition of masking methods and also at the time of computing information loss.

In the context of data privacy protection and evaluation, clustering has an important role. Moreover, it still presents some open research problems and there is room for improvement in existing approaches both in protection and evaluation methods.

Acknowledgements

Partial support by the Spanish MEC (project Co-privacy – TIN2011-27076-C03-03), and MINECO (project MCSec – TIN2014-55243-P) is acknowledged.

References

1. Abril, D., Navarro-Arribas, G., Torra, V., (2010) Towards Semantic Microaggregation of Categorical Data for Confidential Documents, Modeling Decisions for Artificial Intelligence, LNCS 6408, Springer, 266–276.
2. Abril, D., Navarro-Arribas, G., Torra, V., (2015) Spherical microaggregation: Anonymizing sparse vector spaces. Computers & Security 49, 28–44.
3. Batet, M., Erola, A., Snchez, D., Castell-Roca, J. (2014) Semantic Anonymisation of Set-valued Data. ICAART (1) 2014, 102–112.
4. Batet, M., Erola, A., Sánchez D., Castellà-Roca, J., (2013) Utility preserving query log anonymization via semantic microaggregation, Information Sciences, 242, 49–63.
5. Bezdek, J. C. (1981) Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York.
6. Byun, J.-W., Sohn, Y., Bertino, E., Li, N., (2006). Secure Anonymization for Incremental Datasets, Secure Data Management, Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 48–63.
7. Campello, R. J. G. B. (2007) A fuzzy extension of the Rand index and other related indexes for clustering and classification assessment, Pattern Recognition Letters 28:7 833-841
8. Cao, J., Carminati, B., Ferrari, E., Tan, K.-L., (2011) CASTLE: Continuously Anonymizing Data Streams. IEEE Transactions on Dependable and Secure Computing 8, 337–352.
9. De Capitani di Vimercati, S., Foresti, S., Livraga, G., Samarati, P. (2012) Data Privacy: Definitions and Techniques, Int. J. of Unc., Fuzz. and Knowledge Based Systems 20:6 793-817.
10. Defays, D., Nanopoulos, P. (1993), Panels of enterprises and confidentiality: The small aggregates method, Proc. of the 1992 Symposium on Design and Analysis of Longitudinal Surveys, Statistics Canada, pp. 195–204.
11. DMOZ, (2015) The Open Directory Project, www.dmoz.org
12. Domingo-Ferrer, J., González-Nicolás, U. (2010) Hybrid microdata using microaggregation, Information Sciences 180 2834-2844.
13. Domingo-Ferrer, J., Mateo-Sanz, J. M. (2002) Practical data-oriented microaggregation for statistical disclosure control, IEEE Trans. on Knowledge and Data Engineering 14:1 189-201.
14. Domingo-Ferrer, J., Mateo-Sanz, J. M., Torra, V. (2001) Comparing SDC methods for microdata on the basis of information loss and disclosure risk, Pre-proceedings of ETK-NTTS'2001, (Eurostat, ISBN 92-894-1176-5), Vol. 2, 807-826, Creta, Greece.
15. Domingo-Ferrer, J., Torra, V. (2001) Disclosure Control Methods and Information Loss for Microdata, in P. Doyle, J. I. Lane, J. J. M. Theeuwes, L. Zayatz (eds.) Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, Elsevier Science, 91-110.
16. Domingo-Ferrer, J., Torra, V. (2001) A quantitative comparison of disclosure control methods for microdata, in P. Doyle, J. I. Lane, J. J. M. Theeuwes, L. Zayatz (eds.) Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies, North-Holland, 111-134.
17. Domingo-Ferrer, J., Torra, V. (2005) Ordinal, Continuous and Heterogeneous k-Anonymity Through Microaggregation, Data Mining and Knowledge Discovery 11:2 195-212.
18. Erola, A., Castellà-Roca, J., Navarro-Arribas, G., Torra, V., (2011) Semantic microaggregation for the anonymization of query logs using the open directory project, SORT-Statistics and Operations Research Transactions, 41–58. Sep 2011.

19. Fellbaum, C., (1998, ed.) WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.
20. Feder, T., Nabar, S. U., Terzi, E. (2008) Anonymizing graphs, CoRR abs/0810.5578, October.
21. Ghinita, G., Karras, P., Kalnis, P., Mamoulis, N., (2007) Fast data anonymization with low information loss. In: Proceedings of the 33rd International Conference Very Large Data Bases, pp. 758–769.
22. Hansen, S.L., Mukherjee, S., (2003) A polynomial algorithm for optimal univariate microaggregation, IEEE Transactions on Knowledge and Data Engineering, 15(4), 1043–1044.
23. Hay, M., Miklau, G., Jensen, D. (2008) Anonymizing Social Networks, Proc. VLDB 2008.
24. Hore, B., Jammalamadaka, R.C., Mehrotra, S., (2–7) Flexible anonymization for privacy preserving data publishing: a systematic search based approach. In: Proceedings of the 7th SIAM International Conference on Data Mining.
25. Hüllermeier, E., Rifqi, M. (2009) A Fuzzy Variant of the Rand Index for Comparing Clustering Structures, Proc. IFSA-EUSFLAT 2009.
26. Ladra, S., Torra, V. (2008) On the comparison of generic information loss measures and cluster-specific ones, Intl. J. of Unc., Fuzz. and Knowledge-Based Systems, 16:1 107-120.
27. Laszlo, M., Mukherjee, S. (2013) Optimal univariate microaggregation with data suppression, The journal of systems and software 86 677-682.
28. Laszlo, M., Mukherjee, S. (2015) Iterated local search for microaggregation, Journal of Systems and Software 100 15-26
29. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Mondrian multidimensional k-anonymity. In: Proceedings of International Conference on Data Engineering (2006)
30. Li, N., Li, T., Venkatasubramanian, S. (2007) T-closeness: privacy beyond k-anonymity and l-diversity, Proc. of the IEEE ICDE 2007.
31. Liu, J., Wang, K. (2013) Anonymizing bag-valued sparse data by semantic similarity-based clustering, Knowledge and Information Systems 2013 (2012) 435-461.
32. Liu, K., Terzi, E. (2008) Towards identity anonymization on graphs, Proc. SIGMOD 2008.
33. Martínez, S., Sánchez, D., Valls, A., Batet, M. (2012) Privacy protection of textual attributes through a semantic-based masking method. Information Fusion 13(4), 304–314.
34. Martínez, S., Sánchez, D., Valls, A., (2012) Semantic Adaptive Microaggregation of Categorical Microdata. Computers & Security 31(5), 653–672.
35. Miyamoto, S. (1999) Introduction to fuzzy clustering (in Japanese), Ed. Morikita, Japan.
36. Miyamoto, S., Ichihashi, H., Honda, K. (2008) Algorithms for fuzzy clustering, Springer.
37. Navarro-Arribas, G., Abril, D., Torra, V., (2014) Dynamic Anonymous Index for Confidential Data, Data Privacy Management and Autonomous Spontaneous Security. LNCS 8247, Springer, 362–368.
38. Nin, J., Herranz, J., Torra, V. (2008) On the Disclosure Risk of Multivariate Microaggregation, Data and Knowledge Engineering 67 399-412.
39. Nin, J., Herranz, J., Torra, V. (2008) How to Group Attributes in Multivariate Microaggregation, Intl. J. of Unc., Fuzz. and Knowledge-Based Systems, 16:1 121-138.
40. Nin, J., Torra, V. (2009) Analysis of the Univariate Microaggregation Disclosure Risk, New Generation Computing 27 177-194.
41. Oganian, A., Domingo-Ferrer, J., (2001) On the complexity of optimal microaggregation for statistical disclosure control, Statistical Journal of the United Nations Economic Commission for Europe 18(4), 345–353.
42. Pei, J., Xu, J., Wang, Z., Wang, W., Wang, K., (2007) Maintaining K-Anonymity against Incremental Updates, in: 19th International Conference on Scientific and Statistical Database Management, 2007. SSBDM ???07. Presented at the 19th International Conference on Scientific and Statistical Database Management, 2007. SSBDM ???07, pp. 5–5.
43. Samarati, P., (2001) Protecting respondents identities in microdata release. IEEE Transactions on Knowledge and Data Engineering 13, 1010–1027.
44. Solanas, A., Martínez-Balleste, A., Domingo-Ferrer, J., Mateo-Sanz, J.M., (2006). A 2d-tree-based blocking method for microaggregating very large data sets, in: The First International Conference on Availability, Reliability and Security, 2006. ARES 2006.

45. Solanas, A., Pietro, R.D., (2008). A Linear-Time Multivariate Micro-aggregation for Privacy Protection in Uniform Very Large Data Sets, Modeling Decisions for Artificial Intelligence, Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 203–214.
46. Solé, M., Muntés-Mulero, V., Nin, J., (2012). Efficient microaggregation techniques for large numerical data volumes. Int. J. Inf. Secur. 11, 253–267.
47. Stokes, K. (2013) Graph k-Anonymity through k-Means and as Modular Decomposition. Proc. NordSec 2013, LNCS 8208 263-278.
48. Stokes, K., Torra, V. (2012) n-Confusion: a generalization of k-anonymity, Proc. 5th Int. Workshop PAIS.
49. Stokes,K., Torra, V. (2012) Multiple releases of k-anonymous data sets and k-anonymous relational databases, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 20, no. 06, pp. 839–853.
50. Stokes, K., Torra, V. (2011) On some clustering approaches for graphs, Proc. IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011) (ISBN 978-1-4244-7315-1), Taipei, Taiwan, 409-415.
51. Stokes, K., Torra, V. (2012) Reidentification and k-anonymity: a model for disclosure risk in graphs, Soft Computing 16:10 1657-1670.
52. Sweeney, L., (2002). k-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 10, 557–570.
53. Torra, V. (2004), Microaggregation for categorical variables: a median based approach, Proc. Privacy in Statistical Databases (PSD 2004), Lecture Notes in Computer Science 3050, pp. 162–174.
54. Torra, V., Miyamoto, S. (2004) Evaluating fuzzy clustering algorithms for microdata protection, PSD 2004, Lecture Notes in Computer Science 3050 175-186.
55. Torra, V., Narukawa, Y. (2007) Modeling decisions: information fusion and aggregation operators, Springer.
56. Truta, T.M., Campan, A., (2007) K-anonymization incremental maintenance and optimization techniques, Proc. 2007 ACM Symposium on Applied Computing, 380-387.
57. Vaidya, J., Clifton, C., Zhu, M. (2006) Privacy Preserving Data Mining, Springer
58. Xiao, X., Tao, Y., (2007) M-invariance: towards privacy preserving re-publication of dynamic datasets, in: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, SIGMOD ???07. ACM, 689–700.
59. Zhou, B., Pei. J. (2008) Preserving privacy in social networks against neighborhod attacks, Proc. ICDE 2008.