A DENSITY-BASED MICRO AGGREGATION TECHNIQUE FOR PRIVACY-PRESERVING DATA MINING

S. K. Das[1], B. Borah[2]

[1]*Dept. CSE, SMIT, Sikkim*
[2]*Dept. CSE, Tezpur University, Tezpur*

Abstract

Microaggregation is an effective means of protecting the microdata in the statistical databases. Microaggragation protects the microdata by partitioning them into groups of at least k records in each group and substituting the records in each group with the centroid of the group. An optimal microaggregation can be achieved by minimizing the information loss incurred from the aggregation. This paper presents a density-based microagregation method for protecting the numeric data employing the density-based notion of clustering. In this work we provide a microaggragation method which reduces the risk of microdata disclosure and incurs the minimum information loss with incresed data utility. In addition to it the paper also shows an experimental compariosn with the existing heuristics for microaggregation.

Keywords: *Privacy-preserving, microdata, density-based, microaggregation, clustering.*

----------***----------

1. INTRODUCTION

The notion of privacy in the access of information from the different institutes and organizations has attained substantial concern recently as the capability of statistical databases and online storing of information enhanced remarkably. The noble obligation to respect the individual's confidentiality aroused as a major issue while publishing the sensitive information publicly. To come across the purpose several approaches employing different heuristics has been proposed. Microaggregation is a family of the Statistical Disclosure Control (*SDC*) techniques used for protecting sensitive data (microdata) in statistical databases which belongs to the data modification category. A Microaggregation Technique (*MAT*) [1] is used to protect micro-data files by storing the individual records in groups possessing a minimum size constraint. Whenever a query is submitted, it is addressed to a group containing the record, but never to a specific record within the group. This prevents a respondent from isolating a record with overlapping queries. *MAT* holds many attractive features such as its robust performance, its consistent responses, and ease of application. At the same time, the purpose is that there should not be a huge reduction in the information content of the data so that the user is able to get useful, un-biased statistical summaries. Thus, Micro-aggregation can be modeled as a clustering problem with cardinality constraints that are specifically significant to micro-data in the area of *SDC*. In other words, a *MAT* is typically completed by clustering the micro-individual records into groups (where each group satisfies certain group-size constraints *k*) based on the similarity between them, and then replaces the individual values by the aggregated value.

A *microdata* is a set of records containing data of individuals being studied, who can be persons, companies, etc. To obtain microaggregates in a microdata set with *n* data vectors, these are combined to form *g* groups of size at least *k*. The partition problem embedded in microaggregation differs from the classical clustering problem whose goal is to split a population into a fixed number of disjoint groups regardless of the group size. Partitions resulting from microaggregation cannot consist of groups of size smaller than *k*, so we call such partitions as *k-partitions.* Each group contains at least *k* records for a certain *k*. One way to defend the micro data in the databases is to mask and reveal the database that attains *k*-anonymity. A release provide *k*-anonymity protection for $k > 1$ if each other entity in the database is indistinguishable from at least other k-1 entities in the database [2]. In an optimal microaggregation no group has records more than $2k$-1, as the groups having size $\geq 2k$ can be split further to reduce the information loss [3].

For the rest of paper, section 2 presents the background, *DBSCAN* and *MDAV* methods. Section 3 illustrates the proposed method, the *DBM* algorithm. Section 4 provides the experimental results and eventually section 5 presents the conclusions.

2 BACKGROUNDS

2.1 Basic of Microaggregation

As the microaggregation reduces to the clustering problem with a size constraint in each group, hence the homogeneity measure for the records in the groups is significant to quantize the risk of disclosure and information loss. The most common homogeneity measure for clustering is within-group sum of squares errors *SSE* [3]. The within-group homogeneity measure

based on Euclidean distance can be used as a measure of variation within a cluster. If all records within a cluster are identical the *SSE* would then be equal to 0. *SSE* is the sum of squared distances from the centroid of each group to every record in the group. For a *k*-partition, *SSE* can be computed as:

$$\mathbf{SSE} = \sum_{i=1}^{S} \sum_{j=1}^{n_i} (\mathbf{X_{ij}} - \bar{\mathbf{X}}_\mathbf{i})'(\mathbf{X_{ij}} - \bar{\mathbf{X}}_\mathbf{i})$$

Where *S* is the number of groups in the *k*-partition and n_i is the number of records in the *i*-th group. The sum of square distance is computed on the data after standardizing them, by subtracting from the values of each attribute the attribute mean and dividing the result by the attribute standard deviation. The Total Sum of squares, *SST* is defined as the total sum of squares errors within the entire dataset, calculated by aggregating the Euclidian distances of each record X_{ij} to the centroid $\bar{X}$. Here the centroid $\bar{X}$ is the centroid of entire dataset. SST can be calculated as follows.

$$SST = \sum_{i=1}^{S} \sum_{j=1}^{n_i} (\mathbf{X_{ij}} - \bar{\mathbf{X}})'(\mathbf{X_{ij}} - \bar{\mathbf{X}})$$

As the centroid of an attribute is fixed for a dataset so *SST* is fixed for a given dataset regardless of how dataset is partitioned. Information loss is used to quantify the amount of information of a dataset that is lost after microaggregation. The most common definition of information loss (*IL*) [3], is as below:

$$IL = \frac{SSE}{SST}$$

Where *SSE* is the within-group squared errors and *SST* is total sum of squares. We calculate Information Loss as

$$IL = 100 * \frac{SSE}{SST}$$

2.2 Clustering

A cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. The process of grouping a set of physical or abstract objects into classes of similar objects is called clustering. The clustering based on the notion of density [4] used the following notions:

- A central point (*p*)
- A distance metric from the point (*Eps*)
- Minimum number of points within the specified distance metric (*MinPts*)

Eps-neighborhood of a point: For a point *p*, the points contained within the distance metric (*Eps*) is termed as *Eps*-neighborhood of *p* represented as $N_{Eps}(p)$, and defined as:

$$N_{Eps}(p) = \{ q \in D \mid \text{dist}(p, q) \leq Eps \}$$

A point *p* is *directly density-reachable* from a point *q* with respect to *Eps* and *MinPts* if,

1. $p \in N_{Eps}(q)$ and
2. $|N_{Eps}(q)| \geq MinPts$ (core point condition)

A point *p* is *directly density-reachable* (Fig 1) from *q* if there exists a chain of points $p_1, p_{2,\ldots}, p_n$, where $p_1=q$, $p_n=p$, such that p_{i+1} is directly-density reachable from p_i. A point *p* is *density-connected* to point *q* with respect to *Eps* and *MinPts* if there exists a point ***o*** such that both *p* and *q* are density-reachable from ***o*** with respect to *Eps* and *MinPts*.

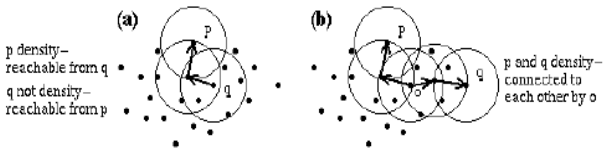

Fig 1: Density-reachability and Density connectivity

Density-connectivity is a symmetric relation. For density reachable points, the relation of density-connectivity is also reflexive. Intuitively, a cluster is defined to be a set of density-connected points which is maximal with respect to density-reachability. Noise will be defined relative to a given set of clusters. Noise is simply the set of points in *D* not belonging to any of its clusters.

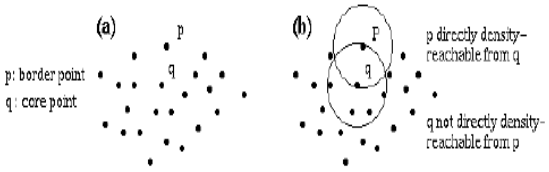

Fig 2: Core point and border point.

A point is a *core* (Fig. 2(a)) point if the number of points within a given neighborhood around a point as determined by the distance function and a user specified parameter, *Eps*, exceeds a certain threshold, *MinPts*. a user-specified parameter. A *border* (Fig. 2(a)) point is not a core point, but it falls within the neighborhood of a core point**.** It is a point which lies on the border of clusters. It has its *Eps-neighborhood* less than *MinPts*. A border point can fall within the neighborhoods of several core points. A *noise* is any point that is neither a core points nor a border point. A cluster *C* with respect to *Eps* and *MinPts* is a non-empty subset of *D* satisfying the following conditions:

1. $\forall p, q$: if $p \in C$ and *q* is density-reachable (Fig. 2(b)) from *p* with respect to *Eps* and *MinPts*, then $q \in$ C. (Maximality)
2. $\forall p, q \in C$, *p* is density-connected to *q* with respect to *Eps* and *MinPts* .(Connecivity)

Let $C_1,.,C_k$ be the clusters of the database D with respect to parameters Eps_i and $MinPts_i$, i =1,..., k. Then we define the *noise* as the set of points in the database D not belonging to any cluster C_i, i.e. $noise = \{ p \in D \mid \forall i: p \notin C_i \}$.

2.3 DBSCAN: Density Based Spatial Clustering of Applications with Noise [4]

The *DBSCAN* (Density Based Spatial Clustering of Applications with Noise) was introduced in [4] by authors Martin Ester, Hans-Peter Kriegel, Jorg Sander, Xiaowei Xu. It was designed to discover clusters and noise in spatial databases. To retrieve all the density reachable points, appropriate values of *Eps* and *MinPts* are to known. *Eps* and *MinPts* values of the "thinnest" cluster are to be found and these values are used globally i.e., *DBSCAN* uses the same values of *Eps* and *MinPts* for all the clusters.

DBSCAN Algorithm:

1. Arbitrarily select a point ***p.***
2. Retrieve all points density-reachable from *p* with respect to *Eps* and *MinPts*.
3. If *p* is a core point, a cluster is formed.
4. If *p* is a border point, no points are density- reachable from *p* and *DBSCAN* visits the next point of the database.
5. Continue the process until all of the points have been processed.

As global values are used for *Eps* and *MinPts*, two clusters C_1 and C_2 can be merged if they are very close to one another. The result of *DBSCAN* is independent of the order in which the points of database are visited except in the following case. If C_1 and C_2 are very close to each other, there may be a point *p* which can belong to both C_1 and C_2. Then, *p* must be a border point in both clusters, if not, and then C_1 and C_2 are same as it uses global parameters. In such case, *p* will be assigned to cluster which is discovered first. The general idea is to continue growing the given cluster as long as the density (number of objects or data points) in the neighborhood exceeds some threshold, that is, for each data point within a given cluster, the neighborhood of a given radius *(Eps*) has to contain at least a minimum number of points *(MinPts).* Clusters are regarded as regions in the data space in which the objects are dense, and which are separated by regions of low object density (noise). These regions may have an arbitrary shape and the points inside a region may be arbitrarily distributed. This method can be used to filter out noise (outliers) and discover clusters of arbitrary shape.

2.4 MDAV

The Maximum Distance to Average Vector method (*MDAV*) is a method for microaggregation to protect the disclosure in statistical databases. *MDAV* was proposed in [5,6] as part of a multivariate microaggregation technique implemented in the μ-*Argus* package for statistical disclosure control. Many heuristics have been proposed in the literature for privacy protection through microaggregation. *MDAV* is a multivariate micro aggregation heuristic, which *k*-partition the dataset, considering the two furthest points in the dataset. It produces the *k*-partition of the data set for the purpose of microaggregation, so in each group attribute value for records is replaced by the average value of records for that particular group. The algorithm is detailed below.

MDAV Algorithm:

1. Compute the centroid (average record) *x* of records in the data set. Find the most distant record *r* from the centroid. Also find the most distant record *s* from *r*.
2. Form two groups around *r* and *s*: the first group contains *r* and *k*-1 records closest to *r*; The other group contains *s* and *k*-1 records closest to *s*.
3. If there are at least 2*k* records which do not belong to any of the groups in Step 2, go to Step 1 taking as new set of points the previous set of records minus the groups formed in the latest instance of Step 2.
4. If there are between *k* and 2*k*-1 record which do not belong to any of the group formed in Step 2, form a new group with those records and exit the algorithm.
5. If there are less than *k* remaining records which do not belong to any of the group formed in Step 2, add them to the group formed in Step 2 whose centroid is closest to the centroid of the remaining points.

3. THE PROPOSED DENSITY-BASED MICROAGGREGATION METHOD (DBM)

As the purpose is to defend the privacy of individual's through microaggregation, our initial requisite is to seek for a *k*-partitioning approach with as more as possible homogeneous records in the same groupin order to incurs less information loss enhancing data utility.

Like other microaggregation methods the proposed *DBM* method microaggregate the records in two successive steps: Partitioning and Aggregation. Initially micro-data file is partitioned into several clusters or groups using density-based notions for clustering. Each of the noise record that does not belong to any cluster is added to the nearest cluster. Some clusters may contain more than 2*k*-1 records. Considering the principle of optimal *k*-partitioning for the optimal microagggregation it further decomposes the large clusters containg more than or equal to 2*k* records using the *MDAV* algorithm so that each cluster contains at least *k* and at most 2*k*-1 records. Then aggregation is done by replacing each record in a cluster by the centroid of the cluster.

The algorithm is detailed below:

Algorithm *DBM* (*Eps*, *k*)

Input: A dataset *D* of *n* records and the value *k* for *k*-partition.

Output: A partitioning, $D := \{P_1, P_2, \ldots, P_n. \mid k \leq |P_{i=1,n}| \leq 2k - 1\}$.

1. Partition dataset *D*, with *DBSCAN*(*Eps*, *k*) as $C = \{ C_1, C_2, \ldots, C_n\}$, such that $C_i \cap C_j = \phi$, $i \neq j$;
   $N := D - \bigcup_{i=0}^{n} C_i$, where *N* the set of noise from *DBSCAN*(*Eps*, *k*);
2. Assign each of the noise points to nearest cluster using *k*-nearest-neighbour;
3. For each, $C_i \in C$, $|C_{i=1,n}| \geq 2k$,
   Call *MDAV*(C_i) to partition $C_i := \{P_{i1}, P_{i2}, \ldots, P_{in}, \mid k \leq |P_{i=1,n}| \leq 2k\text{-}1\}$;
4. End;

The *DBM* algorithm initially forms the cluster with the *DBSCAN*(*Eps*, *k*), for the entire dataset considering all records based on the density-connectivity of each data points. The cluster or groups with the constraints as having number of records more than or equal to 2*k*, where *k* is a user parameter for *k*-anonymization, is further split into clusters having at least *k* records in each with *MDAV*(C_i). The *MDAV* partition the records in C_i, by computing the centroid of the records in C_i and finding the most distant record *r* from the centroid and also finding the most distant record *s* from *r* and then forming two clusters with them with other *k*-1 nearest records around them. It then look for if records left in C_i is between *k* and 2*k*-1, if so it forms a single cluster with them and stops, otherwise if there is less than *k* records left in C_i it adds those records to the nearest cluster formed from C_i and whose centroid is closest to the centroid of the left records in C_i . Thus how *DBM* partition the records for the more similar records in the same cluster satisfying the condition of each cluster having at least *k* record and not more than 2*k*-1 records. The records are then micgraggregated for privacy preservation.

4. EXPERIMENTAL RESULTS

We performed the experiment on the various dataset and measured the Information Loss (*IL*), Sum of Squared Errors (*SSE*) after microaggregating the data. i.e. substituting the confidential attribute by the average of each cluster for the entire partition of the particular dataset.

4.1 Overview of the Datasets:

The three real-world datasets [7] which have been used as benchmarks in prior studies to evaluate various microaggregation heuristics were adopted in our experiments. CENSUS 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 and two additional attributes which are not considered in our experiment. EIA data set obtained from the U.S. Energy Information Authority. It consists of 4092 records with 11 numeric attributes and two categorical attribute (not used in our experiment). TARRAGONA real data set comprises the figures from 834 companies in the area of Tarragona. So the dataset contains 834 records with 13 numeric attributes.

4.2 Data Standardization:

The dataset is first standardized based on the mean and standard deviation of of each attribute. A value, v, of an attribute is standardized to v´ by computing, $v' = (v\text{-}A)/\sigma A$, where A and σ are the mean and standard deviation, respectively, of the values for each attribute.

4.3 Performance Analysis:

4.3.1 Information Loss Comparison

The TABLE 1 provides the comparison of information loss (IL) for the DBM method with different existing methods for Tarragona, Census, and EIA datasets. The minimum measure for each dataset and different methods with same value of k is in boldface. The table above compares DBM with the other existing methods which are, NPN-MHM(Nearest Point Next with Hansen-Mukherjee algorithm) [8], MDAV-MHM [8], CBFS-MHM [8], MD-MHM[8], MD(Maximum Distance) [9], CBFS(Centoid Based Fixed Microaggregation) [10], TFRP-1(Two Fixed Reference Points)[11], TFRP-2 [11], DBA-1(Density-based Algorithm) [12], DBA-2 [12]. It is observed that the proposed method incurs less information loss for Tarragona and Census datasets. DBA2 reduces the information loss the most for the EIA dataset.

Table 1 IL comparison

IL=100*SSE/SST for different values of k, different dataset, different microaggregation heuristics.				
Dataset	Method	K=3	K=5	K=10
Tarragona	NPN-MHM	17.395	27.021	40.183
	MD	16.983	22.527	33.183
	MD-MHM	16.983	22.527	33.183
	MDAV	16.933	22.462	33.193
	MDAV-MHM	16.933	22.462	33.192
	CBFS	16.974	22.828	33.219

	CBFS-MHM	16.971	22.828	33.219
	DBM	9.656	13.046	17.519
Census	NPN-MHM	6.350	11.344	18.734
	MD	5.720	9.006	14.397
	MD-MHM	5.697	8.986	14.397
	MDAV	5.692	9.088	14.224
	MDAV-MHM	5.652	9.087	14.224
	CBFS	5.680	8.905	13.896
	TFRP-1	5.931	9.357	14.442
	TFRP-2	5.803	9.012	13.944
	DBA-1	6.145	10.842	17.785
	DBA-2	5.582	9.046	13.521
	DBM	5.240	8.286	13.317
EIA	NPN-MHM	0.553	0.960	2.319
	MD	0.472	1.669	3.714
	MD-MHM	0.442	1.263	3.637
	MDAV	0.483	1.678	3.845
	MDAV-MHM	0.408	1.256	3.773
	CBFS	0.483	1.748	3.545
	TFRP-1	0.530	1.651	3.242
	TFRP-2	0.428	0.910	2.590
	DBA-1	1.090	1.896	4.266
	DBA-2	0.421	0.818	2.081
	DBM	0.453	1.001	3.236

4.3.2 SSE Comparison

In TABLE 2 we compared the proposed DBM method with SSE value of the three existing method namely MD, MDAV and V-MDAV (Variable-sized Maximum Distance to Average Vector) [13] and observe that proposed method outperforms the other three exiting methods in terms of SSE measure for Census dataset, with k=3,4,5,10 and for EIA dataset with k=3,4, 5.

Table 2 SSE comparison

Data	Method	K=3	K=4	K=5	K=10
Census	MD	803.09	1072.70	1264.51	2021.27
	MDAV	799.18	1053.78	1276.02	1997.03
	V-MDAV	798.49	1055.51	1260.56	1974.75
	DBM	735.74	977.92	1163.36	1869.64
EIA	MD	212.60	347.45	751.44	1671.78
	MDAV	217.38	302.18	750.20	1728.31
	V-MDAV	240.70	337.87	511.20	1270.90
	DBM	203.71	275.83	450.52	1456.55

From the experimental results it can be concluded that DBM provides more homogeneity within the group incurring less error or information loss.

5. CONCLUSIONS AND FUTURE WORKS

Microaggregation is an effective means of protecting the microdata. The optimal microaggregation can be achieved by minimizing the information loss from the aggregation. We presented a microagregation method, DBM algorithm, for microaggregating the numeric data considering density-based notion of clustering. The proposed method microaggregates by k-partitioning the records in two phases. Initially it partitions the records forming groups considering the density-connectivity of each data point in the dataset. Then it adjusts the group size, for each group possessing at least k records. It checks the groups with more than 2k number of records and then split it into smaller groups. The microdata in each group is then replaced with the centroid of that group for the purpose of anonymity. We compared the proposed method against existing different approaches and conclude that it reduces the Information Loss (IL) more than many other existing methods.

The DBM method microaggregate only the numeric microdata as it substitute the values of microdata with an aggregate value to obfuscate and preserve the privacy of sensitive attributes. The categorical data cannot be presented with an aggregate. So to extend the method for categorical and mixed data are to be considered. An approach to determine value of Eps is to be integrated in the proposed method. Feasible enhance of the considered point will enhance the efficiency of the DBM method.

REFERENCES

[1] Charu C. Aggarwal and Philip S. Yu, "Privacy-preserving Data Mining: Models and Algorithms (Advances in Database Systems)", Springer Science and Business Media L.L.C.:Berlin,Heidelberg, 2008.

[2] L. Sweeney, "*k*-anonymity: a model for protecting privacy", International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 10 (5), 2002; 557-570.

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

[4] Martin Ester, Hans-Peter Kriegel, Jörg Sander and Xiaowei Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96).

[5] A. Hundepool, A. V. deWetering, R. Ramaswamy, L. Franconi, A. Capobianchi, P.-P. DeWolf, J. Domingo-Ferrer, V. Torra, R. Brand, and S. Giessing, "μ-ARGUS version 4.0 Software and User's Manual", Voorburg NL: Statistics Netherlands, May 2005, http://neon.vb.cbs.nl/casc.

[6] J. Domingo-Ferrer and V. Torra, "Ordinal, continuous and heterogenerous *k*-anonymity through microaggregation", Data Mining and Knowledge Discovery, vol. 11, no. 2, pp. 195–212, 2005.

[7] J. Domingo-Ferrer, and A. Solanas, "Privacy in Statistical Databases:k-Anonymity Through Microaggregation", *IEEE 2006.*

[8] J. Domingo-Ferrer, A. Martínez-Ballesté, J. M. Mateo-Sanz, and F. Sebé, "Efficient multivariate data-oriented microaggregation", *The VLDB Journal*, 15(4), 355–369. (2006).

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

[10] M. Laszlo and S. Mukherjee, "Minimum spanning tree partitioning algorithm for microaggregation", IEEE Trans. Knowl. Data Eng. 17(7), 902–911, 2005.

[11] C.-C Chang,Y.-C Li and W.-H. Huang, "TFRP: An efficient microaggregation algorithm for statistical disclosure control", Journal of Systems and Software, 80(11), pp. 1866-1878, 2007.

[12] Jun-Lin Lin, Tsung-Hsien Wen, Jui-Chien Hsieh and Pei-Chann Chang, "Density-based microaggregation for statistical disclosure control", Expert Systems with Applications 37 (2010) 3256–3263.

[13] A. Solanas, A. Marteniz-Balleste, "V-MDAV: A multivariate microaggregation with variable group size", In proceedings of the seventh COMPSTAT Symposium of the IASC, Rome, 2006