Scientific Journal Impact Factor (SJIF): 1.711
e-ISSN: 2349-9745
p-ISSN: 2393-8161
International Journal of Modern Trends in Engineering and Research
www.ijmter.com

Cluster Based Access Privilege Management Scheme for Databases

S.Nithya[1], A.Sathyapriya[2], G.Karthik[3]
[1]PG Scholar Computer Science and Engineering, Vivekanandha College Of Engineering For Women
[2]Assistant professor Computer Science and Engineering, Vivekanandha College Of Engineering For Women
[3]Assistant professor Information Technology, Kongunadu College of Engineering and Technology

Abstract-Knowledge discovery is carried out using the data mining techniques. Association rule mining, classification and clustering operations are carried out under data mining. Clustering method is used to group up the records based on the relevancy. Distance or similarity measures are used to estimate the transaction relationship. Census data and medical data are referred as micro data. Data publish schemes are used to provide private data for analysis. Privacy preservation is used to protect private data values. Anonymity is considered in the privacy preservation process.

Data values are allowed to authorized users using the access control models. Privacy Protection Mechanism (PPM) uses suppression and generalization of relational data to anonymize and satisfy privacy needs. Accuracy-constrained privacy-preserving access control framework is used to manage access control in relational database. The access control policies define selection predicates available to roles while the privacy requirement is to satisfy the k-anonymity or l-diversity. Imprecision bound constraint is assigned for each selection predicate. k-anonymous Partitioning with Imprecision Bounds (k-PIB) is used to estimate accuracy and privacy constraints. Role-based Access Control (RBAC) allows defining permissions on objects based on roles in an organization. Top Down Selection Mondrian (TDSM) algorithm is used for query workload-based anonymization. The Top Down Selection Mondrian (TDSM) algorithm is constructed using greedy heuristics and kd-tree model. Query cuts are selected with minimum bounds in Top-Down Heuristic 1 algorithm (TDH1). The query bounds are updated as the partitions are added to the output in Top-Down Heuristic 2 algorithm (TDH2). The cost of reduced precision in the query results is used in Top-Down Heuristic 3 algorithm (TDH3). Repartitioning algorithm is used to reduce the total imprecision for the queries.

The privacy preserved access privilege management scheme is enhanced to provide incremental mining features. Data insert, delete and update operations are connected with the partition management mechanism. Cell level access control is provided with differential privacy method. Dynamic role management model is integrated with the access control policy mechanism for query predicates.

I. Introduction

Privacy-preserving data mining (PPDM) refers to the area of data mining that seeks to safeguard sensitive information from unsolicited or unsanctioned disclosure. Most traditional data mining techniques analyze and model the data set statistically, in aggregation, while privacy preservation is primarily concerned with protecting against disclosure individual data records. This domain separation points to the technical feasibility of PPDM.

Historically, issues related to PPDM were first studied by the national statistical agencies interested in collecting private social and economical data, such as census and tax records and making it available for analysis by public servants, companies and researchers. Building accurate socio economical models is vital for business planning and public policy. Yet, there is no way of knowing in advance what models may be needed, nor is it feasible for the statistical agency to perform all data processing for everyone, playing the role of a trusted third party. Instead, the agency provides the data in a sanitized form that allows statistical processing and protects the privacy of individual records, solving a problem known as privacy-preserving data publishing. For a survey of work in statistical databases, see Adam and Wortmann and Willenborg and de Waal.

The term privacy-preserving data mining was introduced in the papers Agrawal and Srikant and Lindell and Pinkas. These papers considered two fundamental problems of PPDM: privacy-preserving data collection and mining a data set partitioned across several private enterprises. Agrawal and Srikant devised a randomization algorithm that allows a large number of users to contribute their private records for efficient centralized data mining while limiting the disclosure of their values; Lindell and Pinkas invented a cryptographic protocol for decision tree construction over a data set horizontally partitioned between two parties. These methods were subsequently refined and extended by many researchers worldwide. Other areas that influence the development of PPDM include cryptography and secure multiparty computation, database query auditing for disclosure detection and prevention, database privacy and policy enforcement, database security and of course, specific application domains.

II. Related Work

Predicate based fine-grained access control has further been proposed, where user authorization is limited to pre-defined predicates [1]. Enforcement of access control and privacy policies has been studied. Studying the interaction between the access control mechanisms and the privacy protection mechanisms has been missing. Recently, Chaudhuri et al. have studied access control with privacy mechanisms [2]. They use the definition of differential privacy whereby random noise is added to original query results to satisfy privacy constraints. They have not considered the accuracy constraints for permissions. We define the privacy requirement in terms of k-anonymity. It has been shown by Li et al. [6] that after sampling, k-anonymity offers similar privacy guarantees as those of differential privacy. The proposed accuracy-constrained privacy preserving access control framework allows the access control administrator to specify imprecision constraints that the privacy protection mechanism is required to meet along with the privacy requirements.

In our analysis of the related work, we focus on query-aware anonymization. For the state of the art in k-anonymity techniques and algorithms, we refer the reader to a recent survey paper [3]. Workload-aware anonymization is first studied by LeFevre et al. [5] They have proposed the Selection Mondrian algorithm [4], which is a modification to the greedy multidimensional partitioning algorithm Mondrian. In their algorithm, based on the given query-workload, the greedy splitting heuristic minimizes the sum of imprecision for all queries. Iwuchukwu and Naughton have proposed an Rþ-tree based anonymization algorithm. The authors illustrate by experiments that anonymized data using biased Rþ-tree based on the given query workload is more accurate for those queries than for an unbiased algorithm. Ghinita et al. have proposed algorithms based on space filling curves for k-anonymity and l-diversity [7]. They also introduce the problem of accuracy-constrained anonymization for a given bound of acceptable information loss for each equivalence class [8]. Similarly, Xiao et al. [9] propose to add noise to queries according to the size of the queries in a given workload to satisfy differential privacy. Bounds for query imprecision have not been considered. The existing literature on workload-aware anonymization has a focus to minimize the overall imprecision for a given set of queries. Anonymization with imprecision constraints for individual queries has not been studied before. We follow the imprecision definition of LeFevre et al. and introduce the constraint of imprecision bound for each query in a given query workload.

III. Data Privacy Using *k*-Anonymity

Today's globally networked society places great demand on the dissemination and sharing of information, which is probably becoming the most important and demanded resource. While in the past released information was mostly in tabular and statistical form (*macrodata*), many situations call today for the release of specific data (*microdata*). Microdata, in contrast to macrodata reporting precomputed statistics, provide the convenience of allowing the final recipient to perform on them analysis as needed. To protect respondents' identity when releasing microdata, data holders often remove or encrypt explicit identifiers, such as names and social security numbers. De-identifying data, provide no guarantee of anonymity. Released information often contains other data, such as race, birth date, sex and ZIP code that can be linked to publicly available information to re-identify respondents and to infer information that was not intended for release.

One of the emerging concepts in microdata protection is *k-anonymity*, which has been recently proposed as a property that captures the protection of a microdata table with respect to possible re-identification of the respondents to which the data refer. *k*-anonymity demands that every tuple in the microdata table released be indistinguishably related to no fewer than *k* respondents. One of the interesting aspect of *k*-anonymity is its association with protection techniques that preserve the truthfulness of the data.

3.1. Generalization and Suppression

Among the techniques proposed for providing anonymity in the release of microdata, the *k*-anonymity proposal focuses on two techniques in particular: *generalization* and *suppression*, which, unlike other existing techniques, such as scrambling or swapping, preserve the truthfulness of the information. Generalization consists in substituting the values of a given attribute with more general values. To this purpose, the notion of *domain* is extended to capture the generalization process by assuming the existence of a set of *generalized domains*. The set of original domains together with their generalizations is referred to as Dom. Each generalized domain contains generalized values and there exists a mapping between each domain and its generalizations. For instance, ZIP codes can be generalized by dropping, at each generalization step, the least significant digit, postal addresses can be generalized to the street, then to the city, to the county, to the state and so on. This mapping is stated by means of a *generalization relationship* $\leq_D$. Given two domains D_i and $D_j \in$ Dom, $D_i \leq_D D_j$ states that values in domain D_j are generalizations of values in D_i. The generalization relationship $\leq_D$ defines a partial order on the set Dom of domains and is required to satisfy the following conditions:

C1: $\forall Di, Dj, Dz \in$ Dom:

$$D_i \leq_D D_j, D_i \leq_D D_z \rightarrow D_j \leq_D D_z \vee D_z \leq_D D_j$$

C2: all maximal elements of Dom are singleton.

Condition C_1 states that for each domain D_i, the set of domains generalization of D_i is totally ordered and, therefore, each D_i has at most *one* direct generalization domain D_j. It ensures determinism in the generalization process. Condition C2 ensures that all values in each domain can always be generalized to a single value. The definition of a generalization relationship implies the existence, for each domain $D \in$ Dom, of a totally ordered hierarchy, called *domain generalization hierarchy*, denoted DGH_D.

A value generalization relationship, denoted $\leq_V$, associates with each value in domain D_i a unique value in domain D_j, direct generalization of D_i. The value generalization relationship implies the existence, for each domain D, of a *value generalization hierarchy*, denoted VGH_D. It is easy to see that the value generalization hierarchy VGH_D is a *tree*, where the leaves are the values in D and the root is the value in the maximum element in DGH_D. An example of domain and value generalization hierarchies for domains: races (R_0), sex (S_0), a subset of the ZIP codes of San Francisco, USA (Z_0), marital status (M_0) and dates of birth (D_0). The generalization relationship specified for ZIP codes generalizes a 5-digit ZIP code, first to a 4-digit ZIP code and then to a 3-digit ZIP code. The other hierarchies are of immediate interpretation.

Since the approach works on sets of attributes, the generalization relationship and hierarchies are extended to refer to tuples composed of elements of Dom or of their values. Given a domain tuple $DT = <D_1, \ldots\ldots\ldots, D_n>$ such that $D_i \in$ Dom, $i = 1, \ldots\ldots, n$, the domain generalization hierarchy of DT is $DGH_{DT} = DGH_{D1} \times \ldots\ldots \times DGHD_n$, where the Cartesian product is ordered by imposing coordinate-wise order. Since each $DGHD_i$ is totally ordered, DGH_{DT} defines a lattice with DT as its minimal element and the tuple composed of the top of each DGH D_i , $i = 1, \ldots\ldots, n$ as its maximal element. Each path from DT to the unique maximal element of $DGHDT$ defines a possible alternative path, called *generalization strategy*, that can be followed when generalizing a quasi-identifier $QI = \{A_1, \ldots\ldots, A_n\}$ of attributes on domains $D_1, \ldots\ldots, D_n$. For instance, consider domains R_0 (race) and Z_0 (ZIP code) whose generalization hierarchies. The domain generalization hierarchy of the domain tuple $<R_0, Z_0>$ together with the corresponding domain and value generalization strategies. There are three different generalization strategies, corresponding to the three paths from the bottom to the top element of lattice $DGH_{<R0, Z0>}$. Each node of the domain generalization hierarchy corresponds to a generalized table where the attributes in the quasi-identifier have been generalized according the corresponding domain tuple.

IV. Privilege Management Scheme for Databases

Organizations collect and analyze consumer data to improve their services. Access Control Mechanisms (ACM) is used to ensure that only authorized information is available to users. Sensitive information can still be misused by authorized users to compromise the privacy of consumers. The concept of privacy-preservation for sensitive data can require the enforcement of privacy policies or the protection against identity disclosure by satisfying some privacy requirements. In this paper, we investigate privacy-preservation from the anonymity aspect. The sensitive information, even after the removal of identifying attributes, is still susceptible to linking attacks by the authorized users. This problem has been studied extensively in the area of micro data publishing [3] and privacy definitions, e.g., k-anonymity, l-diversity and variance diversity. Anonymization algorithms use suppression and generalization of records to satisfy privacy requirements with minimal distortion of micro data. The anonymity techniques can be used with an access control mechanism to ensure both security and privacy of the sensitive information. The privacy is achieved at the cost of accuracy and imprecision is introduced in the authorized information under an access control policy.

We use the concept of imprecision bound for each permission to define a threshold on the amount of imprecision that can be tolerated. Existing workload aware anonymization techniques [5] minimize the imprecision aggregate for all queries and the imprecision added to each permission/query in the anonymized micro data is not known. Making the privacy requirement more stringent results in additional imprecision for queries. The problem of satisfying accuracy constraints for individual permissions in a policy/workload has not been studied before. The heuristics proposed in this paper for accuracy-constrained privacy-preserving access control are also relevant in the context of workload-aware anonymization. The anonymization for continuous data publishing has been studied in literature [3]. In this paper the focus is on a static relational table that is anonymized only once. To exemplify our approach, role-based access control is assumed. The concept of accuracy constraints for permissions can be applied to any privacy-preserving security policy.

An access control policy that allows the roles to access the tuples under the authorized predicate, e.g., Role CE1 can access tuples under Permission P1. The epidemiologists at the state and county level suggest community containment measures, e.g., isolation or quarantine according to the number of persons infected in case of a flu outbreak. According to the population density in a county, an epidemiologist can advise isolation if the number of persons reported with influenza are greater than 1,000 and quarantine if that number is greater than 3,000 in a single day. The anonymization adds imprecision to the query results and the imprecision bound for each query ensures that the results are within the tolerance required. If the imprecision bounds are not satisfied then unnecessary false alarms are generated due to the high rate of false positives.

The contributions of the paper are as follows. First, we formulate the accuracy and privacy constraints as the problem of k-anonymous Partitioning with Imprecision Bounds (k-PIB) and give hardness results. Second, we introduce the concept of accuracy-constrained privacy-preserving access control for relational data. Third, we propose heuristics to approximate the solution of the k-PIB problem and conduct empirical evaluation.

V. Issues On Privilege Management Scheme

Access Control Mechanisms (ACM) is used to ensure that only authorized information is available to users. Privacy Protection Mechanism (PPM) uses suppression and generalization of relational data to anonymize and satisfy privacy needs. Accuracy-constrained privacy-preserving access control framework is used to manage access control in relational database. The access control policies define selection predicates available to roles while the privacy requirement is to satisfy the k-anonymity or l-diversity. Imprecision bound constraint is assigned for each selection predicate. k-anonymous Partitioning with Imprecision Bounds (k-PIB) is used to estimate accuracy and privacy constraints. Role-based Access Control (RBAC) allows defining permissions on objects based on roles in an organization.

Top Down Selection Mondrian (TDSM) algorithm is used for query workload-based anonymization. The Top Down Selection Mondrian (TDSM) algorithm is constructed using greedy heuristics and kd-tree model. Query cuts

are selected with minimum bounds in Top-Down Heuristic 1 algorithm (TDH1). The query bounds are updated as the partitions are added to the output in Top-Down Heuristic 2 algorithm (TDH2). The cost of reduced precision in the query results is used in Top-Down Heuristic 3 algorithm (TDH3). Repartitioning algorithm is used to reduce the total imprecision for the queries. The following issues are identified from the current privilege management scheme. They are static data based access control model, cell level access control is not supported, imprecision bound estimation is not optimized and fixed access control policy model.

VI. Privacy Preservation Using Clusters

In this section, three algorithms based on greedy heuristics are proposed. All three algorithms are based on kd-tree construction. Starting with the whole tuple space the nodes in the kd-tree are recursively divided till the partition size is between k and 2k. The leaf nodes of the kd-tree are the output partitions that are mapped to equivalence classes. Heuristic 1 and 2 have time complexity of $O(d|Q|^2 n^2)$. Heuristic 3 is a modification over Heuristic 2 to have O(d|Q|nl gn) complexity, which is same as that of TDSM. The proposed query cut can also be used to split partitions using bottom- up (Rþ-tree) techniques.

6.1. Top-Down Heuristic 1 (TDH1)

In TDSM, the partitions are split along the median. Consider a partition that overlaps a query. If the median also falls inside the query then even after splitting the partition, the imprecision for that query will not change as both the new partitions still overlap the query as illustrated. In this heuristic, we propose to split the partition along the query cut and then choose the dimension along which the imprecision is minimum for all queries. If multiple queries overlap a partition, then the query to be used for the cut needs to be selected. The queries having imprecision greater than zero for the partition are sorted based on the imprecision bound and the query with minimum imprecision bound is selected. The intuition behind this decision is that the queries with smaller bounds have lower tolerance for error and such a partition split ensures the decrease in imprecision for the query with the smallest imprecision bound. If no feasible cut satisfying the privacy requirement is found, then the next query in the sorted list is used to check for partition split. If none of the queries allow partition split, then that partition is split along the median and the resulting partitions are added to the output after compaction.

6.2. Top-Down Heuristic 2 (TDH2)

In the Top-Down Heuristic 2 algorithm, the query bounds are updated as the partitions are added to the output. This update is carried out by subtracting the ic Qj Pi value from the imprecision bound BQj of each query, for a Partition, say Pi, that is being added to the output. For example, if a partition of size k has imprecision 5 and 10 for Queries Q_1 and Q2 with imprecision bound 100 and 200, then the bounds are changed to 95 and 190, respectively. The best results are achieved if the kd-tree traversal is depth-first (preorder). Preorder traversal for the kd-tree ensures that a given partition is recursively split till the leaf node is reached. Then, the query bounds are updated. Initially, this approach favors queries with smaller bounds. As more partitions are added to the output, all the queries are treated fairly. During the query bound update, if the imprecision bound for any query gets violated, then that query is put on low priority by replacing the query bound by the query size. The intuition behind this decision is that whatever future partition splits TDH2 makes, the query bound for this query cannot be satisfied. Hence, the focus should be on the remaining queries.

6.3. Top-Down Heuristic 3 (TDH3)

The time complexity of the TDH2 algorithm is $O(d|Q|^2 n^2)$, which is not scalable for large data sets. In the Top-Down Heuristic 3 algorithm (TDH3), we modify TDH2 so that the time complexity of $O(d|Q|n\lg n)$ can be achieved at the cost of reduced precision in the query results. Given a partition, TDH3 checks the query cuts only for the query having the lowest imprecision bound. Also, the second constraint is that the query cuts are feasible only in the case when the size ratio of the resulting partitions is not highly skewed. We use a skew ratio of 1:99 for TDH3 as a threshold. If a query cut results in one partition having a size greater than hundred times the other, then that cut is ignored.

VII. Cluster based Access Privilege Management Scheme for Databases

The privacy preserved access control framework is enhanced to provide incremental mining features. Data insert, delete and update operations are connected with the partition management mechanism. Cell level access control is provided with differential privacy method. Dynamic role management model is integrated with the access control policy mechanism for query predicates. The cluster based access control system is designed with incremental mining mechanism. The system also provides cell level access control mechanism. The system uses the differential privacy to protect cell level access. The system is divided into six major modules. They are data preprocess, role management, query level analysis, clustering process, incremental mining and data retrieval process.

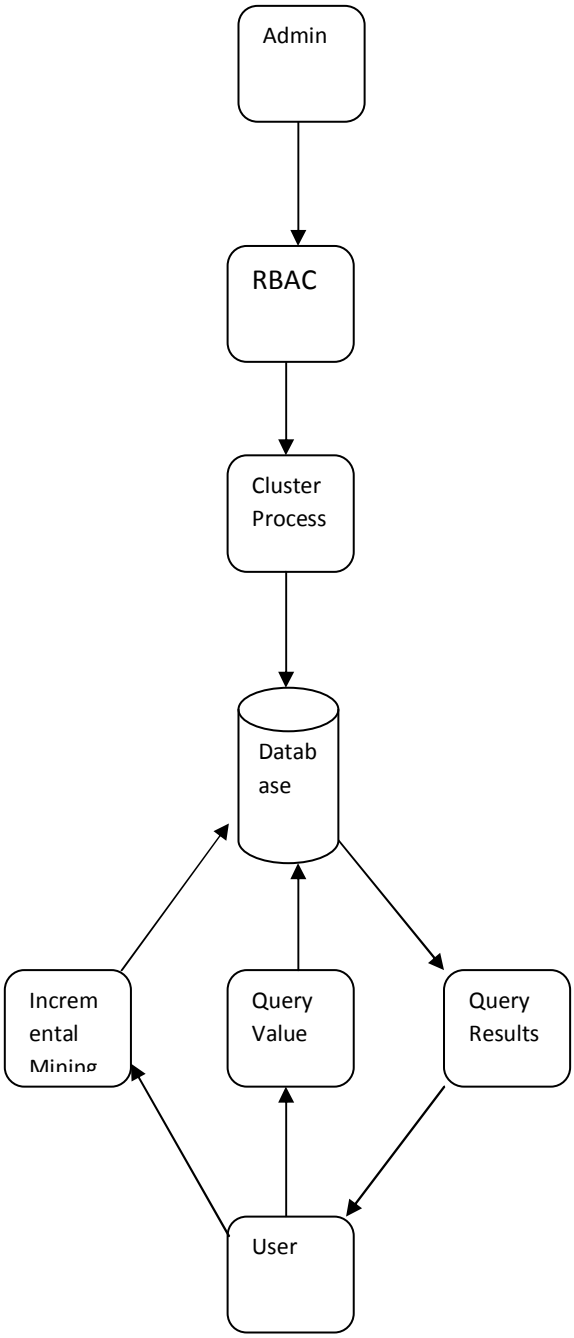

Fig. No: 7.1. Cluster based Access Privilege Management Scheme for Databases

Data preprocess module is designed to perform noise elimination process. User level access permissions are assigned role management process. Query and associated data ranges are analyzed in query level analysis module. Data partitioning is performed in clustering process module. Incremental mining module is designed to modify the database transactions. Data retrieval module is designed to fetch data using query values.

7.1. Data Preprocess

Data populate process is performed to transfer textual data into relational database. Meta data provides the information about the database transactions. Data cleaning process is initiated to correct noisy transactions. Missing values are updated using aggregation based data substitution mechanism.

7.2. Role Management

User details and their access permissions are maintained in the role management process. Sensitive attributes selection is carried out to perform data anonymization process. Each user is assigned with different query values. The query values are used to manage the access permissions to the users.

7.3. Query Level Analysis

User query values are analyzed to estimate the data ranges. Data boundary for each query is estimated using Top-Down Heuristic 1 algorithm (TDH1). TDH2 algorithm is used to update the query bounds as initial partitions. Query results are verified with precision reduction level using TDH3 algorithm.

7.4. Clustering Process

Clustering process is applied to partition the transaction table with query results. TDH based partitioning algorithm is used to cluster the transaction data values. Data partitioning is performed on Anonymized data values. Data partitions are updated into the database.

7.5. Incremental Mining

Data insert, update and delete operations can be performed on the database tables. Tables are associated with the partitioned data values. Reclustering process is performed for the entire database transactions. Cluster refresh process is used to adjust the partitioned data values in incremental mining process.

7.6. Data Retrieval Process

Data retrieval process is carried out using user query values. User query and data retrieval rate are updated into the access logs. User data access is verified with imprecision bound levels. Cell level access control is provided in the query execution process.

VIII. Conclusion

Access control mechanism for relational data is constructed with the privacy preservation based model. Role Based Access Control (RBAC) scheme protects the sensitive data with minimum imprecision values. K-Anonymity model is integrated with minimum imprecision based data access control mechanism. Privacy preserved data access control mechanism is improved with incremental mining model and cell level access control. The system reduces the imprecision rate in query processing. Access control mechanism is adapted for incremental mining model. Time complexity is reduced in the system. The system provides the dynamic policy management mechanism.

REFERENCES

[1] S. Chaudhuri and Sudarshan, "Fine Grained Authorization through Predicated Grants," Proc. IEEE 23rd Int'l Conf. Data Eng.,, 2007.
[2] S. Chaudhuri, Kaushik and R. Ramamurthy, "Database Access Control & Privacy: Is There a Common Ground?" Proc. Fifth Biennial Conf. Innovative Data Systems Research, 2011.
[3] B. Fung, K. Wang, R. Chen and P. Yu, "Privacy-Preserving Data Publishing: A Survey of Recent Developments," ACM Computing Surveys, vol. 42, no. 4, article 14, 2010.
[4] K. LeFevre, DeWitt and R. Ramakrishnan, "Mondrian Multidimensional K-Anonymity," Proc. 22nd Int'l Conf. Data Eng., 2006.
[5] K. LeFevre, D. DeWitt and R. Ramakrishnan, "Workload-Aware Anonymization Techniques for Large-Scale Datasets," ACM Trans. Database Systems, vol. 33, no. 3, pp. 1-47, 2008.
[6] N. Li, W. Qardaji and D. Su, "Provably Private Data Anonymization: Or, k-Anonymity Meets Differential Privacy," Arxiv preprint arXiv:1101.2604, 2011.
[7] G. Ghinita, P. Karras, P. Kalnis and N. Mamoulis, "Fast Data Anonymization with Low Information Loss," Proc. 33rd Int'l Conf. Very Large Data Bases, pp. 758-769, 2007.

[8] G. Ghinita, P. Karras and N. Mamoulis, "A Framework for Efficient Data Anonymization Under Privacy and Accuracy Constraints," ACM Trans. Database Systems, article 9, 2009.
[9] X. Xiao, G. Bender, M. Hay and J. Gehrke, "Ireduct: Differential Privacy with Reduced Relative Errors," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2011.

