Proceedings of SDM'08 International Workshop on

Practical Privacy-Preserving Data Mining

Edited by

Kun Liu
IBM Almaden Research Center, USA

Ran Wolff
University of Haifa, Israel

April 26, 2008
Atlanta, Georgia, USA

Contents

Acknowledgements iv

Message from the Workshop Chairs v
Kun Liu and Ran Wolff

Keynote - Privacy & Data Protection: Policy and Business Trends vi
Harriet P. Pearson

Towards Application-Oriented Data Anonymization 1
Li Xiong, Kumudhavalli Rangachari

Efficient Algorithms for Masking and Finding Quasi-Identifiers 11
Rajeev Motwani, Ying Xu

On the Lindell-Pinkas Secure Computation of Logarithms: From Theory to Practice 21
Raphael S. Ryger, Onur Kardes, Rebecca N. Wright

Constrained k-Anonymity: Privacy with Generalization Boundaries 30
John Miller, Alina Campan, Traian Marius Truta

Privacy-Preserving Predictive Models for Lung Cancer Survival Analysis 40
Glenn Fung, Shipeng Yu, Cary Dehing-Oberije,
Dirk De Ruysscher, Philippe Lambin, Sriram Krishnan, R. Rao Bharat

Acknowledgements

We owe thanks to many people for their contributions to the success of the workshop. We would like to thank first the organizers of SDM'08 for hosting the workshop, and IBM Almaden Research Center for their generous sponsorship. We thank all the authors who submitted papers to the workshop, which allowed us to put together an impressive program.

We would also like to express our thanks to our invited speaker Ms. Harriet P. Pearson, the Chief Privacy Officer of IBM Corporation, for her talk titled "Privacy & Data Protection: Policy and Business Trends".

We would especially like to thank the members of the program committee for giving up their time to review submissions. Despite the short period of time available, they provided thorough, objective evaluations of their allocated papers, ensuring a high standard of presentations at the workshop. Their names are gratefully listed below.

Workshop Co-chairs
Kun Liu and Ran Wolff

Program Committee

Osman Abul, TOBB University, Turkey
Elisa Bertino, Purdue University, USA
Francesco Bonchi, KDD Lab, ISTI-C.N.R., Pisa, Italy
Alexandre Evfimievski, IBM Almaden Research Center, USA
Chris Giannella, Loyola College in Baltimore, Maryland, USA
Murat Kantarcioglu, University of Texas, Dallas, USA
Hillol Kargupta, University of Maryland Baltimore County, USA
Bradley Malin, Vanderbilt University, Nashville, USA
Taneli Mielikäinen, Nokia Research Center Palo Alto, USA
Benny Pinkas, University of Haifa, Israel
Yucel Saygin, Sabanci University, Istanbul, Turkey
Evimaria Terzi, IBM Almaden Research Center, USA
Ke Wang, Simon Fraser University, Canada
Rebecca Wright, Rutgers University, USA
Xintao Wu, University of North Carolina at Charlotte, USA

Sponsored by

SIAM Conference on Data Mining (SDM'08)

IBM Almaden Research Center

Message from the P3DM'08 Workshop Chairs

Kun Liu
IBM Almaden Research Center, USA

Ran Wolff
University of Haifa, Israel

Governmental and commercial organizations today capture large amounts of data on individual behavior and increasingly apply data mining to it. This has raised serious concerns for individuals' civil liberties as well as their economic well being. In 2003, concerns over the U.S. Total Information Awareness (also known as Terrorism Information Awareness) project led to the introduction of a bill in the U.S. Senate that would have banned any data mining programs in the U.S. Department of Defense. Debates over the need for privacy protection vs. service to national security and business interests were held in newspapers, magazines, research articles, television talk shows and elsewhere. Currently, both the public and businesses seem to hold polarized opinions: There are those who think an organization can analyze information it has gathered for any purpose it desires and those who think that every type of data mining should be forbidden. Both positions do little merit to the issue because the former promotes public fear (notably, Sun's Scott McNealy '99 remark "You have no privacy, get over it!") and the latter overly restrictive legislation.

The truth of the matter is not that technology has progressed to the point where privacy is not feasible, but rather the opposite: privacy preservation technology has got to advance to the point where privacy would no longer rely on accidental lack of information but rather on intentional and engineered inability to know. This belief is at the heart of privacy-preserving data mining (PPDM). Pioneered by Agrawal & Srikant and Lindell & Pinkas' work from 2000, there has been an explosive number of publications in this area. Many privacy-preserving data mining techniques have been proposed, questioned, and improved. However, compared with the active and fruitful research in academia, applications of privacy-preserving data mining for real-life problems are quite rare. Without practice, it is feared that research in privacy-preserving data mining will stagnate. Furthermore, lack of practice may hint to serious problems with the underlying concepts of privacy-preserving data mining. Identifying and rectifying these problems must be a top priority for advancing the field.

Following on these understandings, we set out to arrange a workshop on the practical aspects of privacy-preserving data mining. We were encouraged by the enthusiastic response of our PC members, to whom we would like to convey our immense gratitude. The workshop draws eight submissions, of which five were selected for presentation. As you will find in this collection, they range from real PPDM applications to efficiency improvements to known algorithms. Additionally, we pride ourselves in the participation of Harriet P. Pearson, who is the Chief Privacy Officer of IBM. In our perception, CPOs of large businesses such as IBM are likely to be important stake holders in any application of PPDM, and their view should be highly relevant for our community.

Privacy & Data Protection: Policy and Business Trends

Harriet P. Pearson
VP Regulatory Policy & Chief Privacy Officer, IBM Corporation

Business, individuals and the public sector continue to take advantage of the rapid development and adoption of Web-based and other technologies. With innovation in business models and processes, as well as changing individual behaviors in venues such as online social networking, comes the need to address privacy and data protection. This phenomenon occurs every time society has embraced significant new technologies -- whether it be the printing press, telephone or home video rentals. But faster introduction and uptake of technologies in our time results in a larger number of issues and challenges to sort out. This talk will outline the major privacy and data protection trends and their likely effect on the development of public policies, industry practices and technology design.

Towards Application-Oriented Data Anonymization[*†]

Li Xiong[‡] Kumudhavalli Rangachari[§]

Abstract

Data anonymization is of increasing importance for allowing sharing of individual data for a variety of data analysis and mining applications. Most of existing work on data anonymization optimizes the anonymization in terms of data utility typically through one-size-fits-all measures such as data discernibility. Our primary viewpoint in this paper is that each target application may have a unique need of the data and the best way of measuring data utility is based on the analysis task for which the anonymized data will ultimately be used. We take a top-down analysis of typical application scenarios and derive application-oriented anonymization criteria. We propose a prioritized anonymization scheme where we prioritize the attributes for anonymization based on how important and critical they are to the application needs. Finally, we present preliminary results that show the benefits of our approach.

1 Introduction

Data privacy and identity protection is a very important issue in this day and age when huge databases containing a population's information need to be stored and distributed for research or other purposes. For example, the National Cancer Institute initiated the Shared Pathology Informatics Network (SPIN)[1] for researchers throughout the country to share pathology-based data sets annotated with clinical information to discover and validate new diagnostic tests and therapies, and ultimately to improve patient care. However, individually identifiable health information is protected under the Health Insurance Portability and Accountability Act (HIPAA)[2]. The data have to be sufficiently anonymized before being shared over the network.

These scenarios can be generalized into the problem of privacy preserving data publishing where a data custodian needs to distribute an anonymized view of the data that does not contain individually identifiable information to data recipient(s) for various data analysis and mining tasks. Privacy preserving data publishing has been extensively studied in recent years and a few principles have been proposed that serve as criteria for judging whether a published dataset provides sufficient privacy protection [40, 34, 43, 3, 32, 53, 35, 37]. Notably, the earliest principle, k-anonymity [40], requires a set of k records (entities) to be indistinguishable from each other based on a quasi-identifier set, and its extension, l-diversity [34], requires every group to contain at least l well-represented sensitive values. A large body of work contributes to transforming a dataset to meet a privacy principle (dominantly k-anonymity) using techniques such as generalization, suppression (removal), permutation and swapping of certain data values while minimizing certain cost metrics [20, 50, 36, 9, 2, 17, 10, 59, 29, 30, 31, 49, 27, 51, 58].

Most of these methods aim to optimize the data utility measured through a one-size-fitsall cost metric such as general discernibilty or information loss. Few works have considered targeted applications like classification and regression [21, 50, 17, 31] but do not model other kinds of applications nor provide a systematic or adaptive approach for handling various needs.

Contributions. Our primary viewpoint in this paper is that each target application may have a unique need of the data and the best way of measuring data utility is based on the analysis task for which the anonymized data will ultimately be used. We aim to adapt existing methods by incorporating the application needs into the anonymization process, thereby increasing its utility to the target applications.

The paper makes a number of contributions. First, we take a top-down analysis of potential application scenarios and devise models and schemes to represent application requirements in terms of relative attribute

*P3DM'08, April 26, 2008, Atlanta, Georgia, USA.

†This research is partially supported by an Emory URC grant.

‡Dept. of Math & Computer Science, Emory University

§Dept. of Math & Computer Science, Emory University

[1]Shared Pathology Informatics Network. http://www.cancerdiagnosis.nci.nih.gov/spin/

[2]Health Insurance Portability and Accountability Act (HIPAA). http://www.hhs.gov/ocr/hipaa/. State law or institutional policy may differ from the HIPAA standard and should be considered as well.

importance that can be specified by users or learned from targeted analysis and mining tasks. Second, we propose a prioritized anonymization scheme where we prioritize the attributes for anonymization based on how important and critical they are to the application needs. We devise a prioritized cost metric that allows users to assign different weights to different attributes and adapt existing generalization-based anonymization methods in order to produce an optimized view for the user applications. Finally, we present preliminary results that show the benefits of our approach.

2 Related Work

Our research is inspired and informed by a number of related areas. We discuss them briefly below.

Privacy Preserving Access Control and Statistical Databases. Previous work on multilevel secure relational databases [22] provides many valuable insights for designing a fine-grained secure data model. Hippocratic databases [7, 28, 5] incorporate privacy protection within relational database systems. Byun et al. presented a comprehensive approach for privacy preserving access control based on the notion of purpose [14]. While these mechanisms enable multilevel access of sensitive information through access control at a granularity level up to a single attribute value for a single tuple, micro-views of the data are desired where even a single value of a tuple attribute may have different views [13].

Research in statistical databases has focused on enabling queries on aggregate information (e.g. sum, count) from a database without revealing individual records [1]. The approaches can be broadly classified into data perturbation, and query restriction. Data perturbation involves either altering the input databases, or altering query results returned. Query restriction includes schemes that check for possible privacy breaches by keeping audit trails and controlling overlap of successive aggregate queries. The techniques developed have focused only on aggregate queries and relational data types.

Privacy Preserving Data Mining. One data sharing model is the mining-as-a-service model, in which individual data owners submit the data to a data collector for mining or a data custodian outsources mining to an untrusted service provider. The main approach is random perturbation that transforms data by adding random noise in a principled way [8, 48]. The main notion of privacy studied in this context is data uncertainty as versus individual identifiability. There are studies focusing on specific mining tasks such as decision tree [8, 12], association rule mining [39, 15, 16], and on disclosure analysis [26, 19, 42, 12]. A main advantage of data anonymization as opposed to data perturbation is that the released data remain "truthful", though at a coarse level of granularity. This allows various analysis to be carried out using the data, including selection.

Another related area is distributed privacy preserving data sharing and mining that deals with data sharing for specific tasks across multiple data sources in a distributed manner [33, 44, 23, 25, 46, 56, 45, 4, 6, 47, 24, 54, 11, 55]. The main goal is to ensure data is not disclosed among participating parties. Common approaches include data approach that involves data perturbation and protocol approach that applies random-response techniques.

Data Anonymization The work in this paper has its closest roots in data anonymization that provides a micro-view of the data while preserving privacy of individuals. The work in this area can be classified into a number of categories. The first one aims at devising generalization principles in that a generalized table is considered privacy preserving if it satisfies a *generalization principle* [40, 34, 43, 3, 32, 53, 35, 37]. Recent work[52] also considered *personalized anonymity* to guarantee minimum generalization for every individual in the dataset. Another large body of work contributes to the algorithms for transforming a dataset to one that meets a generalization principle and minimizes certain quality metrics. Several hardness results [36, 2] show that computing the optimal generalized table is NP-hard and the result suffers severe information loss when the number of quasi-identifier attributes are high. Optimal solutions [9, 29] enumerate all possible generalized relations with certain constraints using heuristics to prune the search space. Greedy solutions [20, 50, 17, 10, 59, 30, 31, 49] are proposed to obtain a suboptimal solution much faster. A few works are suggesting new approaches in addition to generalization, such as releasing marginals [27], anatomy technique [51], and permutation technique [58], to improve the utility of the published dataset. Another thread of research is focused on disclosure analysis [35]. A few works considered targeted classification and regression applications [20, 50, 17, 31].

Our work builds on top of the existing generalization principles and anonymization techniques and aims to adapt existing solutions for application-oriented anonymization that provides an optimal view for tar-

geted applications.

3 Privacy Model

Among the many identifiability based privacy principles, k-anonymity [41] and its extension l-diversity [34] are the two most widely accepted and serve as the basis for many others, and hence, will be used in our discussions and illustrations. Our work is orthogonal to these privacy principles. Below we introduce some terminologies and illustrate the basic ideas behind these principles.

In defining anonymization, attributes of a given relational table T, are characterized into three types. *Unique identifiers* are attributes that identify individuals. Known identifiers are typically removed entirely from released micro-data. *Quasi-identifier set* is a minimal set of attributes (X_1, ..., X_d) that can be joined with external information to re-identify individual records. We assume that a quasi-identifier is recognized based on domain knowledge. *Sensitive attributes* are those attributes that an adversary should not be permitted to uniquely associate their values with a unique identifier.

Table 1: Illustration of Anonymization: Original Data and Anonymized Data

Name	Age	Gender	Zipcode	Diagnosis
Henry	25	Male	53710	Influenza
Irene	28	Female	53712	Lymphoma
Dan	28	Male	53711	Bronchitis
Erica	26	Female	53712	Influenza

Original Data

Name	Age	Gender	Zipcode	Disease
*	[25 – 28]	Male	[53710-53711]	Influenza
*	[25 – 28]	Female	53712	Lymphoma
*	[25 – 28]	Male	[53710-53711]	Bronchitis
*	[25 – 28]	Female	53712	Influenza

Anonymized Data

Table 1 illustrates an original relational table of personal information. Among the attributes, $Name$ is considered as an identifier, (Age, $Gender$, $Zipcode$) is considered as a quasi-identifer set, and $Diagnosis$ is considered as a sensitive attribute. The k-anonymity model provides an intuitive requirement for privacy in stipulating that no individual record should be uniquely identifiable from a group of k with respect to the quasi-identifier set. The set of all tuples in T containing identical values for the quasi-identifier set X_1, ..., X_d is referred to as an *Equivalence Class*. T is *k-anonymous* with respect to X_1, ..., X_d if every tuple is in an equivalence class of size at least k. A *k-anonymization* of T is a transformation or generalization of the data T such that the transformation is k-anonymous. The l-diversity model provides a natural extension to incorporate a nominal sensitive attribute S. It requires that each equivalence class also contains at least l well-represented distinct values for S. Typical techniques to transform a dataset to satisfy k-anonymity include data generalization, data suppression, and data swapping. Table 1 also illustrates one possible anonymization with respect to a quasi-identifier set ($Age, Gender, Zipcode$) using data generalization that satisfies 2-anonymity and 2-diversity.

4 Application-Oriented Anonymization

Our key hypothesis is that by considering important application requirements, the data anonymization process will achieve a better tradeoff between general data utility and application-specific data utility. We first take a top-down analysis of typical application scenarios and analyze what requirements and implications they pose to the anonymization process. We then present our prioritized optimization metric and anonymization techniques that aim to prioritize the anonymization for individual attributes based on how important they are to target applications.

4.1 Anonymization Goals There are different types of target applications for sharing anonymized data including: 1) query applications supporting ad-hoc queries, 2) applications with a specific mining task such as classification or clustering, and 3) exploratory applications without a specific mining task. We consider two typical scenarios of these applications on anonymized medical data and analyze their implications on the anonymization algorithms.

Scenario 1. Disease-specific public health study. In this study, researchers select a subpopulation of certain health condition (e.g. $Diagnosis =$ "$Lymphoma$"), and study their geographic and demographic distribution, reaction to certain treatment, or survival rate. An example is to identify geographical patterns for the health condition that may be associated with features of the geographic environment.

Scenario 2. Demographic / population study. In this study, researchers may want to study a certain demographic subpopulation (e.g. $Gender = Male$ and

$Age > 50$), and perform exploratory analysis or learn classification models based on demographic information and clinical symptoms to predict diagnosis.

The data analysis for the mentioned applications is typically conducted in two steps: 1) subpopulation identification through a selection predicate, and 2) analysis on the identified subpopulation including mining tasks such as clustering or classification of the population with respect to certain class labels. Given such a two-step process, we identify two requirements for optimizing the anonymization for applications: 1) maximize precision and recall of subpopulation identification, and 2) maximize quality of the analysis.

We first categorize the attributes with respect to the applications on the anonymized data and then explain how the application requirement and optimization goal transform to concrete criteria for application-oriented anonymization. Given an anonymized relational table, each attribute can be characterized by one of the following types with respect to the target applications.

- *Selection attributes* are those attributes used to identify a subpopulation (e.g. *Diagnosis* in Scenario 1 and *Gender* and *Age* in Scenario 2).
- *Feature attributes* are those attributes used to perform analysis such as classifying or clustering data (e.g. *Zipcode* in Scenario 1 for geographic location based analysis).
- *Target attributes* are the class label or attributes for which the classification or prediction are trying to predict (e.g. *Diagnosis* in Scenario 2). Target attributes are not applicable for unsupervised learning tasks such as clustering.

Given the above categorization and the goals in optimizing anonymization for target applications, we derive a set of generalization criteria for the different types of attributes in our anonymization model.

- *Discernibility of selection attributes or predicates.* If a selection attribute is part of the quasi-identifier set and is subject to generalization, it may result in an imprecise query selection. For example, if the *Age* attribute is generalized into ranges of $[0 - 40]$ and [40 *above*], the selection predicate $Age > 50$ in Scenario 2 will result in an imprecise subpopulation. In order to maximize the precision of the population identification, the generalization of the selection attributes should be minimized or adapted to the selection predicates so that the discernibility of selection attributes or predicates are maximized.
- *Discernibility of feature attributes.* For most mining tasks, the anonymized dataset needs to maintain as much information about feature attributes as possible, in order to derive accurate classification models or achieve high quality clustering. As a result, the discernibility of feature attributes needs to be maximized in order to increase data utility.
- *Homogeneity of target attributes.* For classification tasks, an additional criterion is to produce homogeneous partitions or equivalence classes of class labels. The few works specializing on optimizing anonymization for classification applications [21, 50, 17, 31] are mainly focused on this objective. However, it is important to note that if the class label is a sensitive attribute, this criterion is conflicting with the goal of l-diversity and other principles that attempts to achieve a guaranteed level of diversity in sensitive attributes and the question certainly warrants further investigation to achieve best tradeoff.

4.2 Attribute Priorities Based on the above discussion and considering the variety of applications, the first idea we explored is to represent the application requirements using a list of attribute and weight pairs where each attribute is associated with a priority weight based on how important it is to the target applications. We envision that these priority weights can be either explicitly specified by users or implicitly learned by the system based on a set of sample queries and analysis. If the target applications can be fully specified by the users with feature attributes, target attributes, or selection attributes, they can be assigned a higher weight than other attributes in the quasi-identifer set. For instance, in Scenario 1, the attribute-weight list can be represented as $(Age, 0)$, $(Gender, 0)$, $(Zipcode, 1)$ where *Zipcode* is the feature attribute for the location-based study.

Alternatively, the attribute priorities can be learned implicitly from sample queries and analysis. For example, statistics can be collected from query loads on attribute frequencies for projection and selection. In many cases, the attributes in the SELECT clause (projection) correspond to feature attributes while attributes in the WHERE clause (selection) correspond to the selection attributes. The more frequently an attribute is queried,

the more important it is to the application, and the less it should be generalized. Attributes can be then ordered by their frequencies where the weight is a normalized frequency. Another interesting idea is to use a min-term predicate set derived from query load and use that in the anonymization process similar to the data fragmentation techniques in distributed databases. This is on our future research agenda.

4.3 Anonymization Metric Before we can devise algorithms to optimize the solution for the application, we first need to define the optimization objective or the cost function. When the query and analysis semantics are known, a suitable metric for the subpopulation identification process is the *Precision* of the relevant subpopulation similar to the precision of relevant documents in Information Retrieval. Note that a generalized dataset will often produce a larger result set than the original table does with respect to a set of predicates consisting of quasi-identifiers. This is similar to the imprecision metric defined in [31]. For analysis tasks, appropriate metrics for specific analysis tasks should be used as the ultimate optimization goal. This includes accuracy for classification applications and intra-cluster similarity and inter-cluster dissimilarity for clustering applications. The majority metric [25] is a class-aware metric introduced to optimize a dataset for classification applications.

When the query and analysis semantics are not specified, we need a general metric that measures the data utility. Intuitively, the anonymization process should generalize the original data as little as is necessary to satisfy the given privacy principle. There are mainly three cost metrics that have been used in the literature [38], namely, general loss metric, majority metric, and discernibility metric. Among the three, the *discernibility metric*, denoted by C_{DM}, is most commonly used and is defined based on the size of equivalence classes E:

$$C_{DM} = \sum_m |E^m|^2 \tag{4.1}$$

To facilitate the application-oriented anonymization, we devise a prioritized cost metric that allows users to incorporate attribute priorities in order to achieve more granularity for more important attributes. Given a quasi-identifier X_i, let $|E^m_{X_i}|$ denote the size of the mth equivalent class with respect to X_i, let $weight_i$ denote attribute priority associated with attribute X_i, the metric is defined as follows:

$$C_{WDM} = \sum_i weight_i * \sum_m |E^m_{X_i}|^2 \tag{4.2}$$

Consider our example Scenario 1, if given an anonymized dataset such as in Table 1, the discernibility of equivalent classes along attribute *Zipcode* will be penalized more than the other two attributes because of the importance of geographic location. This metric corresponds well with our weighted attributed list representation of the application requirements. It provides a general judgement of the anonymization for exploratory analysis when there is some knowledge about attribute importance in applications but not sufficient knowledge about specific subpopulation or applications.

4.4 Anonymization A large number of algorithms have been developed for privacy preserving data anonymization. They can be roughly classified into top-down and bottom-up approaches and single dimensional and multidimensional approaches. Most of the techniques take a greedy approach and rely on certain heuristics at each step or iteration for selecting an attribute for partitioning (top-down) or generalization (bottom-up). In this study, we adapt the greedy top-down Mondrian multidimensional approach [30] and investigate heuristics for adapting it based on our prioritized optimization metric. It is on our future research agenda to explore various anonymization approaches and investigate systematic ways for adapting them towards application-oriented anonymization.

The Mondrian algorithm (based on k-anonymity principle) uses greedy recursive partitioning of the (multidimensional) quasi-identifer domain space. In order to obtain approximately uniform partition occupancy, it recursively chooses the split attribute with the largest normalized range of values, referred to as *spread*, and (for continuous or ordinal attributes) partitions the data around the median value of the split attribute. This process is repeated until no allowable split remains, meaning that a particular region cannot be further divided without violating the anonymity constraint, or constraints imposed by value generalization hierarchies.

The key of the algorithm is to select the best attribute for splitting (partitioning) during each iteration. In addition to using the spread (range) of the values of

each attribute i, denoted as $spread_i$, in the original algorithm, our approach explores additional metrics.

Attribute priority. Since our main generalization criteria is to maximize the discernibility of important attributes including selection attributes, feature attributes and class attributes for target applications, we use the attribute priority weight for attribute i, denoted by $weight_i$, as an important selection criteria. Attributes with a larger weight will be selected for partitioning so that important attributes will have a more precise view in the anonymized data.

Information gain. When target applications are well specified a priori, another important generalization criterion for classification applications is to maximize the homogeneity of class attributes within each equivalence class. This is reminiscent of decision tree construction where each path of the decision tree leads to a homogeneous group of class labels [18]. Similarly, information gain can be used as a scoring metric for selecting the best attribute for partitioning in order to produce equivalence classes of homogeneous class labels. The information gain for a given attribute i, denoted by $infogain_i$, is computed as the weighted entropy of the resultant partitions based on the split of attribute i:

$$infogain_i = \sum_{P'} \left(\frac{|P'|}{|P|} \sum_{c \in D_c} -p(c|P')logp(c|P')\right) \quad (4.3)$$

where P denotes the current partition, P' denotes the set of resultant partitions of the iteration, $p(c|P')$ is the fraction of tuples in P' with class label c, and D_c is the domain of the class variable c.

The attribute selection criteria for each iteration selects the best attribute based on an overall scoring metric determined by an aggregation of the above metrics. In this study, we use a linear combination of the individual metrics, denoted by O_i for attribute i:

$$O_i = \sum_j (w_j * metric_i^j) / \sum_j w_j \quad (4.4)$$

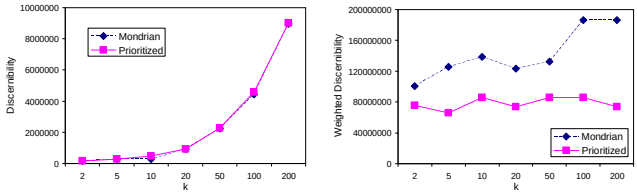
where $metric_i^j \in \{spread_i, infogain_i, weight_i\}$, and w_j is the weight of the individual metric j ($w_j >= 0$).

5 Experiments

We performed a set of preliminary experiments evaluating our approach. The main questions we would like to answer are: 1) does the prioritized anonymization metric (weighted discernibility metric) correlate with good data utility from applications point of view? 2) does the prioritized anonymization scheme provide better data utility than general approaches?

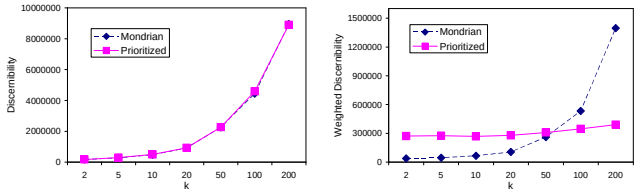
We implemented a prioritized anonymization algorithm based on the Mondrian algorithm [30]. It uses a combined heuristic of the spread and attribute priorities (without information gain) and aims to minimize the prioritized cost metric (instead of the general discernibility metric). We conducted two sets of experiments for exploratory and classification applications respectively.

5.1 Exploratory Applications For exploratory applications, we used the Adults dataset from UC Irvine Machine Learning Repository configured as in [30]. We considered a simple application scenario that requires precise information on a single demographic attribute (*Age* and *Sex* respectively) and hence it is assigned with a higher weight than other attributes in the experiment. The dataset were anonymized using the Mondrian and prioritized approach respectively and we compare the weighted discernibility as well as general discernibility of the two anonymized datasets.



Discernibility Weighted Discernibility

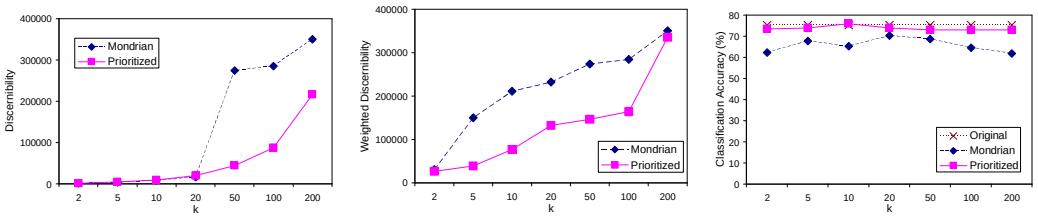
Figure 1: Adult Dataset (*Sex*-Prioritized)



Discernibility Weighted Discernibility

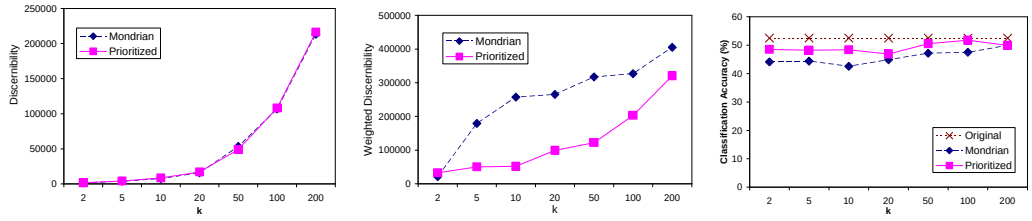
Figure 2: Adult Dataset (*Age*-Prioritized)

Figure 1 and 2 compare the prioritized approach and



(a) Discernibility (b) Weighted Discernibility (c) Classification Accuracy

Figure 3: Japanese Credit Screening Dataset - Classification



(a) Discernibility (b) Weighted Discernibility (c) Prediction Accuracy

Figure 4: Japanese Credit Screening Dataset - Prediction (A3)

the Mondrian approach in terms of general discernibility and weighted discernibility with respect to different value of k for Sex-prioritized and Age-prioritized anonymization respectively. We observe that even though the prioritized approach has a comparable general discernibility with the Mondrian, it achieves a much improved weighted discernibility in both cases, which is directly correlated with the user-desired data utility (i.e. having a more fine-grained view for Age attribute or Sex attribute for exploratory query or mining purposes).

5.2 Classification Applications For classification applications, we used the Japanese Credit Screening dataset, also from the UCI Machine Learning Repository. The dataset consists of 653 instances, 15 attributes and a 2-valued class attribute ($A16$) that corresponds to a positive/negative (+/-) credit. The missing valued instances were removed and the experiments were carried out considering only the continuous attributes ($A2, A3, A8, A11, A14$ and $A15$). The dataset was anonymized using the prioritized approach and the Mondrian approach and the resultant anonymized data as well as the original data were used for classification and prediction. The Weka implementation of the simple Naive-Bayes classifier was used for the classification, with 10 fold cross-validation for classification accuracy determination.

For classification, the class attribute was recoded as 1.0/0.0. Different feature attributes were selected and given varying weights (both arbitrary or assuming user knowledge) to examine their effect on classification accuracy. For prediction, attributes other than the class attribute were recoded into ranges using equi-width[3] approach. A target attribute is selected as the prediction attribute and the rest of the attributes are anonymized and used to predict the target attribute.

We assume the users have some domain knowledge of which attributes will be used as feature attributes for their classification and we then assigned higher priority weights for these attributes. In addition, we also experimented with a set of single-attribute classification by selecting one feature attribute each time and assigned weights for the attributes based on their classification accuracy. The results are similar and we report the first set of results below.

Figure 3(a) and 3(b) compare the prioritized and Mondrian approach in terms of general discernibility and weighted discernibility of the anonymized dataset respectively. Figure 3(c) compares the anonymized datasets as well as the original dataset in terms of accuracy for the class attribute. Similarly, Figure 4 presents the results for prediction of attribute $A3$. We observe that the prioritized approach performs better than the Mondrian for both classification and prediction in terms of accuracy and achieves a comparable accuracy as the original dataset. In addition, a comparison of the dis-

[3]Equal spread ranges for the recoded attributes.

cernibility metrics and the classification accuracy shows that the weighted discernibility metric corresponds well to the application-oriented data utility, i.e. the classification accuracy.

6 Conclusion and Discussions

We presented an application-oriented approach for data anonymization that takes into account the relative attribute importance for target applications. We derived a set of generalization criteria for application-oriented data anonymization and presented a prioritized generalization approach that aims to minimize the prioritized cost metric. Our initial results show that the prioritized anonymization metric correlates well with application-oriented data utility and the prioritized approach achieves better data utility than general approaches from application point of view.

There are a few items on our research agenda. First, the presented anonymization technique uses a special generalization algorithm and a simple weighted heuristic. We will study different heuristics and generalize the result to more advanced privacy principles and anonymization approaches. Second, while it is not always possible for users to specify the attribute priorities before hand, we will study how to automatically learn attribute priorities from sample queries and mining tasks and further devise models and presentations that allow application requirements to be incorporated. In addition, a more in-depth and longer-term issue that we will investigate is the notion of priorities, in particular, the interaction between what data owners perceive and what the data users (applications) perceive. Finally, it is important to note that there are inference implications of releasing multiple anonymized views where multiple data users may collude and combine their views to breach data privacy. While there is work beginning investigating the inference problem [57], the direction certainly warrants further research.

References

[1] N. R. Adams and J. C. Wortman. Security-control methods for statistical databases: a comparative study. *ACM Computing Surveys*, 21(4), 1989.

[2] C. C. Aggarwal. On k-anonymity and the curse of dimensionality. In *VLDB*, pages 901–909, 2005.

[3] G. Aggarwal, T. Feder, K. Kenthapadi, S. Khuller, R. Panigrahy, D. Thomas, and A. Zhu. Achieving anonymity via clustering. In *PODS*, pages 153–162, 2006.

[4] G. Aggarwal, N. Mishra, and B. Pinkas. Secure computation of the kth ranked element. In *IACR Conference on Eurocrypt*, 2004.

[5] R. Agrawal, P. Bird, T. Grandison, J. Kieman, S. Logan, and W. Rjaibi. Extending relational database systems to automatically enforce privacy policies. In *21st ICDE*, 2005.

[6] R. Agrawal, A. Evfimievski, and R. Srikant. Information sharing across private databases. In *SIGMOD*, 2003.

[7] R. Agrawal, J. Kieman, R. Srikant, and Y. Xu. Hippocratic databases. In *VLDB*, 2002.

[8] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proc. of the ACM SIGMOD Conference on Management of Data*, pages 439–450. ACM Press, May 2000.

[9] R. J. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. In *ICDE '05: Proceedings of the 21st International Conference on Data Engineering (ICDE'05)*, pages 217–228, Washington, DC, USA, 2005. IEEE Computer Society.

[10] E. Bertino, B. Ooi, Y. Yang, and R. H. Deng. Privacy and ownership preserving of outsourced medical data. In *ICDE*, 2005.

[11] S. S. Bhowmick, L. Gruenwald, M. Iwaihara, and S. Chatvichienchai. Private-iye: A framework for privacy preserving data integration. In *ICDE Workshops*, page 91, 2006.

[12] S. Bu, L. V. S. Lakshmanan, R. T. Ng, and G. Ramesh. Preservation of patterns and input-output privacy. In *ICDE*, pages 696–705, 2007.

[13] J. Byun and E. Bertino. Micro-views, or on how to protect privacy while enhancing data usability - concept and challenges. *SIGMOD Record*, 35(1), 2006.

[14] J.-W. Byun, E. Bertino, and N. Li. Purpose based access control of complex data for privacy protection. In *ACM Symposium on Access Control Models and Technologies (SACMAT)*, 2005.

[15] A. V. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *PODS*, pages 211–222, 2003.

[16] A. V. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. *Inf. Syst.*, 29(4):343–364, 2004.

[17] B. C. M. Fung, K. Wang, and P. S. Yu. Top-down specialization for information and privacy preservation. In *Proc. of the 21st IEEE International Conference on Data Engineering (ICDE 2005)*, pages 205–216, Tokyo, Japan, April 2005.

[18] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2nd edition, 2006.

[19] Z. Huang, W. Du, and B. Chen. Deriving private information from randomized data. In *SIGMOD Conference*, pages 37–48, 2005.

[20] V. S. Iyengar. Transforming data to satisfy privacy constraints. In *KDD*, pages 279–288, 2002.

[21] V. S. Iyengar. Transforming data to satisfy privacy constraints. In *SIGKDD*, 2002.

[22] S. Jajodia and R. Sandhu. Toward a multilevel secure relational data model. In *ACM SIGMOD*, 1991.

[23] M. Kantarcioglu and C. Clifton. Privacy preserving data mining of association rules on horizontally partitioned data. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 16(9), 2004.

[24] M. Kantarcioglu and C. Clifton. Privacy preserving k-nn classifier. In *ICDE*, 2005.

[25] M. Kantarcoglu and J. Vaidya. Privacy preserving naive bayes classifier for horizontally partitioned data. In *IEEE ICDM Workshop on Privacy Preserving Data Mining*, 2003.

[26] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. On the privacy preserving properties of random data perturbation techniques. In *ICDM*, pages 99–106, 2003.

[27] D. Kifer and J. Gehrke. Injecting utility into anonymized datasets. In *SIGMOD Conference*, pages 217–228, 2006.

[28] K. LeFevre, R. Agrawal, V. Ercegovac, R. Ramakrishnan, Y. Xu, and D. DeWitt. Limiting disclosure in hippocratic databases. In *30th International Conference on Very Large Data Bases*, 2004.

[29] K. LeFevre, D. Dewitt, and R. Ramakrishnan. Incognito: Efficient full-domain k-anonymity. In *ACM SIGMOD International Conference on Management of Data*, 2005.

[30] K. LeFevre, D. DeWitt, and R. Ramakrishnan. Mondrian multidimensional k-anonymity. In *IEEE ICDE*, 2006.

[31] K. LeFevre, D. DeWitt, and R. Ramakrishnan. Workload-aware anonymization. In *SIGKDD*, 2006.

[32] N. Li and T. Li. t-closeness: Privacy beyond k-anonymity and l-diversity. In *To appear in International Conference on Data Engineering (ICDE)*, 2007.

[33] Y. Lindell and B. Pinkas. Privacy preserving data mining. *Journal of Cryptology*, 15(3), 2002.

[34] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *Proceedings of the 22nd International Conference on Data Engineering (ICDE'06)*, page 24, 2006.

[35] D. J. Martin, D. Kifer, A. Machanavajjhala, J. Gehrke, and J. Y. Halpern. Worst-case background knowledge for privacy-preserving data publishing. In *ICDE*, pages 126–135, 2007.

[36] A. Meyerson and R. Williams. On the complexity of optimal k-anonymity. In *PODS*, pages 223–228, 2004.

[37] M. E. Nergiz, M. Atzori, and C. Clifton. Hiding the presence of individuals from shared databases. In *SIGMOD Conference*, pages 665–676, 2007.

[38] M. E. Nergiz and C. Clifton. Thoughts on k-anonymization. In *ICDE Workshops*, page 96, 2006.

[39] S. Rizvi and J. R. Haritsa. Maintaining data privacy in association rule mining. In *VLDB*, pages 682–693, 2002.

[40] L. Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):557–570, 2002.

[41] L. Sweeney. k-anonymity: a model for protecting privacy. *International journal on uncertainty, fuzziness and knowledge-based systems*, 10(5), 2002.

[42] Z. Teng and W. Du. Comparisons of k-anonymization and randomization schemes under linking attacks. In *ICDM*, pages 1091–1096, 2006.

[43] T. M. Truta and B. Vinay. Privacy protection: p-sensitive k-anonymity property. In *ICDE Workshops*, page 94, 2006.

[44] J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *ACM SIGKDD*, 2002.

[45] J. vaidya and C. Clifton. Privacy-preserving k-means clustering over vertically partitioned data. In *SIGKDD*, 2003.

[46] J. Vaidya and C. Clifton. Privacy preserving nave bayes classifier for vertically partitioned data. In *ACM SIGKDD*, 2003.

[47] J. Vaidya and C. Clifton. Privacy-preserving top-k queries. In *ICDE*, 2005.

[48] V. S. Verykios, E. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin, and Y. Theodoridis. State-of-the-art in privacy preserving data mining. *ACM SIGMOD Record*, 33(1), 2004.

[49] K. Wang and B. C. M. Fung. Anonymizing sequential releases. In *ACM SIGKDD*, 2006.

[50] K. Wang, P. S. Yu, and S. Chakraborty. Bottom-up generalization: a data mining solution to privacy protection. In *Proc. of the 4th IEEE International Conference on Data Mining (ICDM 2004)*, November 2004.

[51] X. Xiao and Y. Tao. Anatomy: Simple and effective privacy preservation. In *VLDB*, pages 139–150, 2006.

[52] X. Xiao and Y. Tao. Personalized privacy preservation. In *SIGMOD '06: Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, 2006.

[53] X. Xiao and Y. Tao. M-invariance: towards privacy preserving re-publication of dynamic datasets. In *SIGMOD Conference*, pages 689–700, 2007.

[54] L. Xiong, S. Chitti, and L. Liu. Topk queries across multiple private databases. In *25th International Conference on Distributed Computing Systems (ICDCS 2005)*, 2005.

[55] L. Xiong, S. Chitti, and L. Liu. Mining multiple private databases using a knn classifier. In *ACM Symposium of Applied Computing (SAC)*, pages 435–440, 2007.

[56] Z. Yang, S. Zhong, and R. N. Wright. Privacy-preserving classification of customer data without loss of accuracy. In *SIAM SDM*, 2005.

[57] C. Yao, X. S. Wang, and S. Jajodia. Checking for k-anonymity violation by views. In *VLDB*, 2005.

[58] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu. Aggregate query answering on anonymized tables. In *ICDE*, pages 116–125, 2007.

[59] S. Zhong, Z. Yang, and R. N. Wright. Privacy-enhancing k-anonymization of customer data. In *PODS*, 2005.

Efficient Algorithms for Masking and Finding Quasi-Identifiers *

Rajeev Motwani † Ying Xu ‡

Abstract

A quasi-identifier refers to a subset of attributes that can uniquely identify most tuples in a table. Incautious publication of quasi-identifiers will lead to privacy leakage. In this paper we consider the problems of finding and masking quasi-identifiers. Both problems are provably hard with severe time and space requirements. We focus on designing efficient approximation algorithms for large data sets.

We first propose two natural measures for quantifying quasi-identifiers: distinct ratio and separation ratio. We develop efficient algorithms that find small quasi-identifiers with provable size and separation/distinct ratio guarantees, with space and time requirements sublinear in the number of tuples. We also propose efficient algorithms for masking quasi-identifiers, where we use a random sampling technique to greatly reduce the space and time requirements, without much sacrifice in the quality of the results. Our algorithms for masking and finding quasi-identifiers naturally apply to stream databases. Extensive experimental results on real world data sets confirm efficiency and accuracy of our algorithms.

1 Introduction

A quasi-identifier (also called a semi-key) is a subset of attributes which uniquely identifies most entities in the real world or tuples in a table. A well-known example is that the combination of gender, date of birth, and zipcode can uniquely determine about 87% of the population in United States. Quasi-identifiers play an important role in many aspects of data management, including privacy, data cleaning, and query optimization.

As pointed out in the seminal paper of Sweeney [25], publishing data with quasi-identifiers leaves open attacks that combine the data with other publicly available information to identify represented individuals. To avoid such linking attacks via quasi-identifiers, the concept of k-anonymity was proposed [25, 24] and many algorithms for k-anonymity have been developed [23, 2, 4]. In this paper we consider the problem of masking quasi-identifiers: we want to publish a subset of attributes (we either publish the exact value of every tuple on an attribute, or not publish the attribute at all), so that no quasi-identifier is revealed in the published data. This can be viewed as a variant of k-anonymity where the suppression is only allowed at the attribute level. While this approach is admittedly too restrictive in some applications, there are two reasons we consider it. First, the traditional tuple-level suppression may distort the distribution of the original data and the association between attributes, so sometimes it might be desirable to publish fewer attributes with complete and accurate information. Second, as noted in [15], the traditional k-anonymity algorithms are expensive and do not scale well to large data sets; by restricting the suppression to a coarser level we are able to design more efficient algorithms.

We also consider the problem of finding small keys and quasi-identifiers, which can be used by adversaries to perform linking attacks. When a table which is not properly anonymized is published, an adversary would be interested in finding keys or quasi-identifiers in the table so that once he collects other persons' information on those attributes, he will be able to link the records to real world entities. Collecting information on each attribute incurs certain cost to the adversary (for example, he needs to look up yellow pages to collect the area code of phone numbers, to get party affiliation information from the voter list, etc), so the adversary wishes to find a subset of attributes with a small size or weight that is a key or almost a key to minimize the attack cost.

Finding quasi-identifiers also has other important applications besides privacy. One application is data cleaning. Integration of heterogeneous databases sometimes causes the same real-world entity to be represented by multiple records in the integrated database due to spelling mistakes, inconsistent conventions, etc. A critical task in data cleaning is to identify and remove such fuzzy duplicates [3, 6]. We can estimate the ratio of fuzzy duplicates, for example by checking some samples manually or plotting the distribution of pairwise similarity; now if we can find a quasi-

*P3DM'08, April 26, 2008, Atlanta, Georgia, USA.

†Stanford University. rajeev@cs.stanford.edu. Supported in part by NSF Grant ITR-0331640, and a grant from Media-X.

‡Stanford University. xuying@cs.stanford.edu. Supported in part by Stanford Graduate Fellowship and NSF Grant ITR-0331640.

identifier whose "quasiness" is similar to the fuzzy duplicate ratio, then those tuples which collide on the quasi-identifier are likely to be fuzzy duplicates. Finally, quasi-identifiers are a special case of approximate functional dependency [13, 22], and their automatic discovery is valuable to query optimization and indexing [9].

In this paper, we study the problems of finding and masking quasi-identifiers in given tables. Both problems are provably hard with severe time and space requirements, so we focus on designing efficient approximation algorithms for large data sets. First we define measures for quantifying the "quasiness" of quasi-identifiers. We propose two natural measures – separation ratio and distinct ratio.

Then we consider the problem of finding the minimum key. The problem is NP-hard and the best-known approximation algorithm is a greedy algorithm with approximation ratio $O(\ln n)$ (n is the number of tuples); however, even this greedy algorithm requires multiple scans of the table, which are expensive for large databases that cannot reside in main memory and prohibitive for stream databases. To enable more efficient algorithms, we sacrifice accuracy by allowing approximate answers (quasi-identifiers). We develop efficient algorithms that find small quasi-identifiers with provable size and separation/distinct ratio guarantees, with both space and time complexities sublinear in the number of input tuples.

Finally we present efficient algorithms for masking quasi-identifiers. We use a random sampling technique to greatly reduce the space and time requirements, without sacrificing much in the quality of the results.

Our algorithms for masking and finding minimum quasi-identifiers naturally apply to stream databases: we only require one pass over the table to get a random sample of the tuples and the space complexity is sublinear in the number of input tuples (at the cost of only providing approximate solutions).

1.1 Definitions and Overview of Results

A *key* is a subset of attributes that uniquely identifies each tuple in a table. A *quasi-identifier* is a subset of attributes that can distinguish almost all tuples. We propose two natural measures for quantifying a quasi-identifier. Since keys are a special case of functional dependencies, our measures for quasi-identifiers also conform with the measures of approximate functional dependencies proposed in earlier work [13, 22, 11, 8].

(1) An *α-distinct quasi-identifier* is a subset of attributes which becomes a key in the table remaining after the removal of at most a $1-\alpha$ fraction of tuples in the original table.

(2) We say that a subset of attributes *separates* a pair of tuples x and y if x and y have different values on at least one attribute in the subset. An *α-separation quasi-identifier* is a subset of attributes which separates at least an α fraction of all possible tuple pairs.

	age	sex	state
1	20	Female	CA
2	30	Female	CA
3	40	Female	TX
4	20	Male	NY
5	40	Male	CA

Table 1. An example table. The first column labels the tuples for future references and is not part of the table.

We illustrate the notions with an example (Table 1). The example table has 3 attributes. The attribute *age* is a 0.6-distinct quasi-identifier because it has 3 distinct values in a total of 5 tuples; it is a 0.8-separation quasi-identifier because 8 out of 10 tuple pairs can be separated by *age*. $\{sex, state\}$ is 0.8-distinct and 0.9-separation.

The separation ratio of a quasi-identifier is always larger than its distinct ratio, but there is no one-to-one mapping. Let us consider a 0.5-distinct quasi-identifier in a table of 100 tuples. One possible scenario is that projected on the quasi-identifier there are 50 distinct values and each value corresponds to 2 tuples, so its separation ratio is $1-\frac{50}{\binom{100}{2}} \approx 0.99$; another possible scenario is that for 49 of the 50 distinct values there is only one tuple for each value, and all the other 51 tuples have the same value, and then this quasi-identifier is 0.75-separation. Indeed, an α-distinct quasi-identifier can be an α'-separation quasi-identifier where α' can be as small as $2\alpha-\alpha^2$, or as large as $1-\frac{2(1-\alpha)}{n}$. Both distinct ratio and separation ratio are very natural measures for quasi-identifiers and have different applications as noted in the literature on approximate functional dependency. In this paper we study quasi-identifiers using both measures.

Given a table with n tuples and m attributes, we consider the following problems. The *size* of a key (quasi-identifier) refers to the number of attributes in the key.

Minimum Key Problem: find a key of the minimum size. This problem is provably hard so we also consider its relaxed version:

(ϵ, δ)-Separation or -Distinct Minimum Key Problem: look for a quasi-identifier with a small size such that, with probability at least $1-\delta$, the output quasi-identifier has separation or distinct ratio at least $1-\epsilon$.

β-Separation or -Distinct Quasi-identifier Masking Problem: delete a minimum number of attributes such that there is no quasi-identifier with separation or distinct ratio greater than β in the remaining attributes.

In the example of Table 1, $\{age, state\}$ is a minimum key, with size 2; the optimal solution to 0.8-distinct quasi-identifier masking problem is $\{sex, state\}$; the optimal solution to 0.8-separation quasi-identifier masking problem is $\{age\}$, $\{sex\}$ or $\{state\}$, all of size 1.

The result data after quasi-identifier masking can be viewed as an approximation to k-anonymity. For example, after 0.2-distinct quasi-identifier masking, the result data is approximately 5-anonymous, in the sense that on average each tuple is indistinguishable from another 4 tuples. It does not provide perfect privacy as there may still exist some tuple with a unique value, nevertheless it provides anonymity for the majority of the tuples. The k-anonymity problem is NP-hard [17, 2]; further, Lodha and Thomas [15] note that there is no efficient approximation algorithm known that scale well for large data sets, and they also aim at preserving privacy for majority. We hope to provide scalable anonymizing algorithm by relaxing the privacy constraints. Finally we would like to maximize the utility of published data, and we measure utility in terms of the number of attributes published (our solution can be generalized to the case where attributes have different weights and utility is the weighted sum of published attributes).

We summarize below the contributions of this paper.

1. We propose greedy algorithms for the (ϵ, δ)-separation and distinct minimum key problems, which find small quasi-identifiers with provable size and separation (distinct) ratio guarantees, with space and time requirements sublinear in n. In particular, the space complexity is $O(m^2)$ for the (ϵ, δ)-separation minimum key problem, and $O(m\sqrt{mn})$ for (ϵ, δ)-distinct. The algorithms are particularly useful when $n \gg m$, which is typical of database applications where a large table may consist of millions of tuples, but only a relatively small number of attributes. We also extend the algorithms to find the approximate minimum β-separation quasi-identifiers. (Section 2)
2. We present greedy algorithms for β-separation and β-distinct quasi-identifier masking. The algorithms are slow on large data sets, and we use a random sampling technique to greatly reduce the space and time requirements, without much sacrifice in the utility of the published data. (Section 3)
3. We have implemented all the above algorithms and conducted extensive experiments using real data sets. The experimental results confirm the efficiency and accuracy of our algorithms. (Section 4)

2 Finding Minimum Keys

In this section we consider the Minimum Key problem. First we show the problem is NP-hard (Section 2.1) and the best approximation algorithm is a greedy algorithm which gives $O(\ln n)$-approximate solution (Section 2.2). The greedy algorithm requires multiple scans of the table, which is expensive for large tables and inhibitive for stream databases. To enable more efficient algorithms, we relax the problem by allowing approximate answers, i.e. the (ϵ, δ)-Separation (Distinct) Minimum Key problem. We develop random sampling based algorithms with approximation guarantees and sublinear space (Section 2.3, 2.4).

2.1 Hardness Result

The Minimum Key problem is NP-Hard, which follows easily from the NP-hardness of the *Minimum Test Collection* problem.

> *Minimum Test Collection:* Given a set S of elements and a collection C of subsets of S, a test collection is a subcollection of C such that for each pair of distinct elements there is some set that contains exactly one of the two elements. The Minimum Test Collection problem is to find a test collection with the smallest cardinality.

Minimum Test Collection is equivalent to a special case of the Minimum Key problem where each attribute is boolean: let S be the set of tuples and C be all the attributes; each subset in C corresponds to an attribute and contains all the tuples whose values are *true* in this attribute, then a test collection is equivalent to a key in the table. Minimum Test Collection is known to be NP-hard [7], therefore the Minimum Key problem is also NP-hard.

2.2 A Greedy Approximation Algorithm

The best known approximation algorithm for Minimum Test Collection is a greedy algorithm with approximation ratio $1 + 2\ln|S|$ [18], i.e. it finds a test collection with size at most $1 + 2\ln|S|$ times the smallest test collection size. The algorithm can be extended to the more general Minimum Key problem, where each attribute can be from an arbitrary domain, not just boolean.

Before presenting the algorithm, let us consider a naive greedy algorithm: compute the separation (or distinct) ratio of each attribute in advance; each time pick the attribute with the highest separation ratio in the remaining attributes, until we get a key. The algorithm is fast and easy to implement, but unfortunately it does not perform well when the attributes are correlated. For example if there are many attributes pairwise highly correlated and each has a high separation ratio, then the optimal solution probably includes only one of these attributes while the above greedy algorithm is likely to pick all of them. The approximation ratio of this algorithm can be arbitrarily bad.

A fix to the naive algorithm is to pick each time the attribute which separates the largest number of tuple pairs not yet separated. To prove the approximation ratio of the algorithm, we reduce Minimum Key to the Minimum Set Cover problem. The reduction plays an important role

in designing algorithms for finding and masking quasi-identifiers in later sections.

> *Minimum Set Cover:* Given a finite set S (called the *ground set*) and a collection C of subsets of S, a *set cover* I is a subcollection of C such that every element in S belongs to at least one member of I. *Minimum Set Cover* problem asks for a set cover with the smallest size.

Given an instance of Minimum Key with n tuples and m attributes, we reduce it to a set cover instance as follows: the ground set S consists of all distinct unordered pairs of tuples ($|S| = \binom{n}{2}$); each attribute c in the table is mapped to a subset containing all pairs of tuples separated by attribute c. Now a collection of subsets covers S if and only if the corresponding attributes can separate all pairs of tuples, i.e., those attributes form a key, therefore there is a one-to-one map between minimum set covers and minimum keys.

Consider the example of Table 1. The ground set of the corresponding set cover instance contains 10 elements where each element is a pair of tuples. The column *age* is mapped to a subset c_{age} with 8 pairs: $\{(1,2),(1,3),(1,5),(2,3),(2,4),(2,5),(3,4),(4,5)\}$; the column *sex* is mapped to a subset c_{sex} with 6 pairs, and *state* 7 pairs. The attribute set $\{age, sex\}$ is a key; correspondingly the collection $\{c_{age}, c_{sex}\}$ is a set cover.

The *Greedy Set Cover Algorithm* starts with an empty collection (of subsets) and adds subsets one by one until every element in S has been covered; each time it chooses the subset covering the largest number of uncovered elements. It is well known that this greedy algorithm is a $1 + \ln |S|$ approximation algorithm for Minimum Set Cover.

LEMMA 2.1. *[12] The Greedy Set Cover Algorithm outputs a set cover of size at most $1 + \ln |S|$ times the minimum set cover size.*

The *Greedy Minimum Key Algorithm* mimics the greedy set cover algorithm: start with an empty set of attributes and add attributes one by one until all tuple pairs are separated; each time chooses an attribute separating the largest number of tuple pairs not yet separated. The running time of the algorithm is $O(m^3 n)$. It is easy to infer the approximation ratio of this algorithm from Lemma 2.1:

THEOREM 2.1. *Greedy Minimum Key Algorithm outputs a key of size at most $1 + 2 \ln n$ times the minimum key size.*

The greedy algorithms are optimal because neither problem is approximable within $c \ln |S|$ for some $c > 0$ [10]. Note that this is the worst case bound and in practice the algorithms usually find much smaller set covers or keys.

2.3 (ϵ, δ)-Separation Minimum Key

The greedy algorithm in the last section is optimal in terms of approximation ratio, however, it requires multiple scans ($O(m^2)$ scans indeed) of the table, which is expensive for large data sets. In this and next section, we relax the minimum key problem by allowing quasi-identifiers and design efficient algorithms with approximate guarantees.

We use the standard (ϵ, δ) formulation: with probability at least $1 - \delta$, we allow an "error" of at most ϵ, i.e. we output a quasi-identifier with separation (distinct) ratio at least $1 - \epsilon$. The (ϵ, δ) Minimum Set Cover Problem is defined similarly and requires the output set cover covering at least a $1 - \epsilon$ fraction of all elements.

Our algorithms are based on random sampling. We first randomly sample k elements (tuples), and reduce the input set cover (key) instance to a smaller set cover (key) instance containing only the sampled elements (tuples). We then solve the exact minimum set cover (key) problem in the smaller instance (which is again a hard problem but has much smaller size, so we can afford to apply the greedy algorithms in Section 2.2), and output the solution as an approximate solution to the original problem. The number of samples k is carefully chosen so that the error probability is bounded. We present in detail the algorithm for (ϵ, δ)-set cover in Section 2.3.1; the (ϵ, δ)-Separation Minimum Key problem can be solved by reducing to (ϵ, δ) Minimum Set Cover (Section 2.3); we discuss (ϵ, δ)-Distinct Minimum Key in Section 2.4.

2.3.1 (ϵ, δ) Minimum Set Cover

The key observation underlying our algorithm is that to check whether a given collection of subsets is a set cover, we only need to check some randomly sampled elements if we allow approximate solutions. If the collection only covers part of S, then it will fail the check after enough random samples. The idea is formalized as the following lemma.

LEMMA 2.2. *$s_1, s_2, \ldots, s_k$ are k elements independently randomly chosen from S. If a subset S' satisfies $|S'| < \alpha |S|$, then $Pr[s_i \in S', \forall i] < \alpha^k$.*

The proof is straightforward. The probability that a random element of S belongs to S' is $|S'|/|S| < \alpha$, therefore the probability of all k random elements belonging to S' is at most α^k.

Now we combine the idea of random sample checking with the greedy algorithm for the exact set cover. Our *Greedy Approximate Set Cover algorithm* is as follows:

1. Choose k elements uniformly at random from S (k is defined later);
2. Reduce the problem to a smaller set cover instance: the ground set $\tilde{S}$ consists of the k chosen elements; each subset in the original problem maps to a subset which is the intersection of $\tilde{S}$ and the original subset;
3. Apply Greedy Set Cover Algorithm to find an exact set cover for $\tilde{S}$, and output the solution as an approximate set cover to S.

Let n be the size of the ground set S, and m be the number of subsets. We say a collection of subsets is an *α-set cover* if it covers at least an α fraction of the elements.

THEOREM 2.2. *With probability $1-\delta$, the above algorithm with $k = \log_{\frac{1}{1-\epsilon}} \frac{2^m}{\delta}$ outputs a $(1-\epsilon)$-set cover whose cardinality is at most $(1 + \ln\log_{\frac{1}{1-\epsilon}} \frac{2^m}{\delta})|I^*|$, where I^* is the optimal exact set cover.*

Proof. Denote by $\tilde{S}$ the ground set of the reduced instance ($|\tilde{S}| = k$); by $\tilde{I}^*$ the minimum set cover of $\tilde{S}$. The greedy algorithm outputs a subcollection of subsets covering all k elements of $\tilde{S}$, denoted by $\tilde{I}$. By Lemma 2.1, $|\tilde{I}| \leq (1 + \ln|\tilde{S}|)|\tilde{I}^*|$. Note that I^*, the minimum set cover of the original set S, corresponds to a set cover of $\tilde{S}$, so $|\tilde{I}^*| \leq |I^*|$, and hence $|\tilde{I}| \leq (1 + \ln k)|I^*|$.

We map $\tilde{I}$ back to a subcollection I of the original problem. We have

$|I| = |\tilde{I}| \leq (1 + \ln k)|I^*| = (1 + \ln\log_{\frac{1}{1-\epsilon}} \frac{2^m}{\delta})|I^*|$.

Now bound the probability that I is not a $1-\epsilon$-set cover. By Lemma 2.2, the probability that a subcollection covering less than a $1-\epsilon$ fraction of S covers all k chosen elements of $\tilde{S}$ is at most

$$(1-\epsilon)^k = (1-\epsilon)^{\log_{\frac{1}{1-\epsilon}} \frac{2^m}{\delta}} = (1-\epsilon)^{\log_{1-\epsilon} \frac{\delta}{2^m}} = \frac{\delta}{2^m}.$$

There are 2^m possible subcollections; by union bound, the overall error probability, i.e. the probability that any subcollection is not a $(1-\epsilon)$-cover of S but is an exact cover of $\tilde{S}$, is at most δ. Hence, with probability at least $1-\delta$, I is a $(1-\epsilon)$-set cover for S.

If we take ϵ and δ as constants, the approximation ratio is essentially $\ln m + O(1)$, which is smaller than $1 + \ln n$ when $n \gg m$. The space requirement of the above algorithm is $mk = O(m^2)$ and running time is $O(m^4)$.

2.3.2 (ϵ, δ)-Separation Minimum Key The reduction from Minimum Key to Minimum Set Cover preserves the separation ratio: an α-separation quasi-identifier separates at least an α fraction of all pairs of tuples, so its corresponding subcollection is an α-set cover; and vice versa. Therefore, we can reduce the (ϵ, δ)-Separation Minimum Key problem to the (ϵ, δ)-Set Cover problem where $|S| = O(n^2)$. The complete algorithm is as follows.

1. Randomly choose $k = \log_{\frac{1}{1-\epsilon}} \frac{2^m}{\delta}$ pairs of tuples;
2. Reduce the problem to a set cover instance where the ground set $\tilde{S}$ is the set of those k pairs and each attribute maps to a subset of the k pairs separated by this attribute;
3. Apply Greedy Set Cover Algorithm to find an exact set cover for $\tilde{S}$, and output the corresponding attributes as a quasi-identifier to the original table.

THEOREM 2.3. *With probability $1-\delta$, the above algorithm outputs a $(1-\epsilon)$-separation quasi-identifier whose size is at most $(1 + \ln\log_{\frac{1}{1-\epsilon}} \frac{2^m}{\delta})|I^*|$, where I^* is the smallest key.*

The proof directly follows Theorem 2.2. The approximation ratio is essentially $\ln m + O(1)$. The space requirement of the above algorithm is $mk = O(m^2)$, which significantly improves upon the input size mn.

2.4 (ϵ, δ)-Distinct Minimum Key

Unfortunately, the reduction to set cover does not necessarily map an α-distinct quasi-identifier to an α-set cover. As pointed out in Section 1.1, an α-distinct quasi-identifier corresponds to an α'-separation quasi-identifier, and thus reduces to an α'-set cover, where α' can be as small as $2\alpha - \alpha^2$, or as large as $1 - \frac{2(1-\alpha)}{n}$. Therefore reducing this problem directly to set cover gives too loose bound, and a new algorithm is desired.

Our algorithm for finding distinct quasi-identifiers is again based on random sampling. We reduce the input (ϵ, δ)-Distinct Minimum Key instance to a smaller (exact) Minimum Key instance by randomly choosing k tuples and keeping all m attributes. The following lemma bounds the probability that a subset of attributes is an (exact) key in the sample table, but not an α-distinct quasi-identifier in the original table.

LEMMA 2.3. *Randomly choose k tuples from input table T to form table T_1. Let p be the probability that an (exact) key of T_1 is not an α-distinct quasi-identifier in T. Then*

$$p < e^{-\frac{(\frac{1}{\alpha}-1)k(k-1)}{2n}}$$

Proof: Suppose we have n balls distributed in $d = \alpha n$ distinct bins. Randomly choose k balls without replacement, and the probability that the k balls are all from different bins is exactly p. Let $x_1, x_2, \ldots, x_d$ be the number of balls in the d bins ($\sum_{i=1}^{d} x_i = n, x_i > 0$), then

$$p = \frac{\sum_{all\{i_1,i_2,\ldots,i_k\}} x_{i_1} x_{i_2} \ldots x_{i_k}}{\binom{n}{k}}.$$

p is maximized when all x_is are equal, i.e. each bin has $\frac{1}{\alpha}$ balls. Next we compute p for this case. The first ball can be from any bin; to choose the second ball, we have $n-1$ choices, but it cannot be from the same bin as the first one, so $\frac{1}{\alpha} - 1$ of the $n-1$ choices are infeasible; similar arguments hold for the remaining balls. Summing up, the probability that all k balls are from distinct bins is

$$\begin{aligned} p &= 1(1 - \frac{\frac{1}{\alpha}-1}{n-1})(1 - \frac{2(\frac{1}{\alpha}-1)}{n-2}) \ldots (1 - \frac{(k-1)(\frac{1}{\alpha}-1)}{n-(k-1)}) \\ &\leq e^{-(\frac{\frac{1}{\alpha}-1}{n-1} + \frac{2(\frac{1}{\alpha}-1)}{n-2} + \frac{(k-1)(\frac{1}{\alpha}-1)}{n-(k-1)})} \\ &< e^{-\frac{(\frac{1}{\alpha}-1)k(k-1)}{2n}} \square \end{aligned}$$

The Greedy (ϵ, δ)-Distinct Minimum Key Algorithm is as follows:

1. Randomly choose $k = \sqrt{\frac{2(1-\epsilon)}{\epsilon} n \ln \frac{2^m}{\delta}}$ tuples and keep all attributes to form table T_1;
2. Apply Greedy Minimum Key Algorithm to find an exact key in T_1, and output it as a quasi-identifier to the original table.

THEOREM 2.4. *With probability $1-\delta$, the above algorithm outputs a $(1-\epsilon)$-distinct quasi-identifier whose size is at most $(1+2\ln k)|I^*|$, where I^* is the smallest exact key.*

The proof is similar to Theorem 2.2, substituting Lemma 2.2 with Lemma 2.3. k is chosen such that $p \leq \frac{\delta}{2^m}$ to guarantee that the overall error probability is less than δ. The approximation ratio is essentially $\ln m + \ln n + O(1)$, which improves the $1 + 2\ln n$ result for the exact key. The space requirement is $mk = O(m\sqrt{mn})$, sublinear in the number of tuples of the original table.

2.5 Minimum β-Separation Quasi-identifier

In previous sections, our goal is to find a small quasi-identifier that is almost a key. Note that ϵ indicates our "error tolerance", not our goal. For (ϵ, δ)-Separation Minimum Key problem, our algorithm is likely to output quasi-identifiers whose separation ratios are far greater than $1 - \epsilon$. For example, suppose the minimum key of a given table consists of 100 attributes, while the minimum 0.9-separation quasi-identifier has 10 attributes, then our $(0.1, 0.01)$-separation algorithm may output a quasi-identifier that has say 98 attributes and is 0.999-separation. However, sometimes we may be interested in finding 0.9-separation quasi-identifiers which have much smaller sizes. For this purpose we consider the *Minimum β-Separation Quasi-identifier Problem*: find a quasi-identifier with the minimum size and separation ratio at least β.

The Minimum β-Separation Quasi-identifier Problem is at least as hard as Minimum Key since the latter is a special case where $\beta = 1$. So again we consider the approximate version by relaxing the separation ratio: we require the algorithm to output a quasi-identifier with separation ratio at least $(1-\epsilon)\beta$ with probability at least $1 - \delta$.

We present the algorithm for approximate β-set cover; the β-separation quasi-identifier problem can be reduced to β-set cover as before.

The *Greedy Minimum β-Set Cover algorithm* works as follows: first randomly sample $k = \frac{16}{\beta\epsilon^2} \ln \frac{2^m}{\delta}$ elements from the ground set S, and construct a smaller set cover instance defined on the k chosen elements; run the greedy algorithm on the smaller set cover instance until get a subcollection covering at least $(2-\epsilon)\beta k/2$ elements (start with an empty subcollection; each time add to the subcollection a subset covering the largest number of uncovered elements).

THEOREM 2.5. *The Greedy Minimum β-Set Cover algorithm runs in space $mk = O(m^2)$, and with probability at least $1 - \delta$, outputs a $(1-\epsilon)\beta$-set cover with size at most $(1 + \ln \frac{(2-\epsilon)\beta k}{2})|I^*|$, where I^* is the minimum β-set cover of S.*

The proof can be found in our technical report. This algorithm also applies to the minimum exact set cover problem (the special case where $\beta = 1$), but the bound is worse than Theorem 2.2; see our technical report for detailed comparison.

The minimum β-separation quasi-identifier problem can be solved by reducing to β-set cover problem and applying the above greedy algorithm. Unfortunately, we cannot provide similar algorithms for β-distinct quasi-identifiers; the main difficulty is that it is hard to give a tight bound to the distinct ratio of the original table by only looking at a small sample of tuples. The negative results on distinct ratio estimation can be found in [5].

3 Masking Quasi-Identifiers

In this section we consider the quasi-identifier masking problem: when we release a table, we want to publish a subset of the attributes subject to the privacy constraint that no β-separation (or β-distinct) quasi-identifier is published; on the other hand we want to maximize the utility, which is measured by the number of published attributes. For each problem, we first present a greedy algorithm which generates good results but runs slow for large tables, and then show how to accelerate the algorithms using random sampling. (The algorithms can be easily extended to the case where the attributes have weights and the utility is the sum of attribute weights.)

3.1 Masking β-Separation Quasi-identifiers

As in Section 2.2, we can reduce the problem to a set cover type problem: let the ground set S be the set of all pairs of tuples, and let each attribute correspond to a subset of tuple pairs separated by this attribute, then the problem of Masking β-Separation Quasi-identifier is equivalent to finding a maximum number of subsets such that at most a β fraction of elements in S is covered by the selected subsets. We refer to this problem as *Maximum Non-Set Cover problem*. Unfortunately, the Maximum Non-Set Cover problem is NP-hard by a reduction from the Dense Subgraph problem. (See our technical report for the hardness proof.)

We propose a greedy heuristic for masking β-separation quasi-identifiers: start with an empty set of attributes, and add attributes to the set one by one as long as the separation ratio is below β; each time pick the attribute separating the least number of tuple pairs not yet separated.

The algorithm produces a subset of attributes satisfying the privacy constraint and with good utility in practice,

however it suffers from the same efficiency issue as the greedy algorithm in Section 2.2: it requires $O(m^2)$ scans of the table and is thus slow for large data sets. We again use random sampling technique to accelerate the algorithm: the following lemma gives a necessary condition for a β-separation quasi-identifier in the sample table (with high probability), so only looking at the sample table and pruning all attribute sets satisfying the necessary condition will guarantee the privacy constraint. The proof of the lemma is omit for lack of space.

LEMMA 3.1. *Randomly sample k pairs of tuples, then a β-separation quasi-identifier separates at least $\alpha\beta$ of the k pairs, with probability at least $1 - e^{-(1-\alpha)^2\beta k/2}$.*

The *Greedy Approximate β-Separation Masking Algorithm* is as follows:

1. Randomly choose k pairs of tuples;
2. Let $\beta' = (1 - \sqrt{\frac{2\ln(2^m/\delta)}{\beta k}})\beta$. Run the following greedy algorithm on the selected pairs: start with an empty set C of attributes, and add attributes to the set C one by one as long as the number of separated pairs is below $\beta' k$; each time pick the attribute separating the least number of tuple pairs not yet separated;
3. Publish the set of attributes C.

By the nature of the algorithm the published attributes C do not contain quasi-identifiers with separation greater than β' in the sample pairs; by Lemma 3.1, this ensures that with probability at least $1 - 2^m e^{-(1-\beta'/\beta)^2\beta k/2} = 1 - \delta$, C do not contain any β-separation quasi-identifier in the original table. Therefore the attributes published by the above algorithm satisfies the privacy constraint.

THEOREM 3.1. *With probability at least $1 - \delta$, the above algorithm outputs an attribute set with separation ratio at most β.*

We may over-prune because the condition in Lemma 3.1 is not a sufficient condition, which means we may lose some utility. The parameter k in the algorithm offers a tradeoff between the time/space complexity and the utility. Obviously both the running time and the space increase linearly with k; on the other hand, the utility (the number of published attributes) also increases with k because the pruning condition becomes tighter as k increases. Our experiment results show that the algorithm is able to dramatically reduce the running time and space complexity, without much sacrifice in the utility (see Section 4).

3.2 Masking β-Distinct Quasi-identifiers

For masking β-distinct quasi-identifiers, we can use a similar greedy heuristic: start with an empty set of attributes, and each time pick the attribute adding the least number of distinct values, as long as the distinct ratio is below β. And similarly we can use a sample table to trade off utility for efficiency.

1. Randomly choose k tuples and keep all the columns to form a sample table T_1;
2. Let $\beta' = (1 - \sqrt{\frac{2\ln(2^m/\delta)}{\beta k}})\beta$. Run the following greedy algorithm on T_1: start with an empty set C of attributes, and add attributes to the set C one by one as long as the distinct ratio is below β'; each time pick the attribute adding the least number of distinct values;
3. Publish the set of attributes C.

Lemma 3.2 and Theorem 3.2 state the privacy guarantee of the above algorithm.

LEMMA 3.2. *Randomly sample k tuples from the input table T into a small table T_1 ($k \ll n$, where n is the number of tuples in T). A β-distinct quasi-identifier of T is an $\alpha\beta$-distinct quasi-identifier of T_1 with probability at least $1 - e^{-(1-\alpha)^2\beta k/2}$.*

Proof. By the definition of β-distinct quasi-identifier, the tuples has at least βn distinct values projected on the quasi-identifier. Take (any) one tuple from each distinct value, and call those representing tuples "good tuples". There are at least βn good tuples in T.

Let k_1 be the number of distinct values in T_1 projected on the quasi-identifier, and k' be the number of good tuples in T_1. We have $k_1 \geq k'$ because all good tuples are distinct. (The probability that any good tuple is chosen more than once is negligible when $k \ll n$.) Next we bound the probability $Pr[k' \leq \alpha\beta k]$. Since each random tuple has a probability at least β of being good, and each sample are chosen independently, we can use Chernoff bound (see [19] Ch. 4) and get

$$Pr[k' \leq \alpha\beta k] \leq e^{-(1-\alpha)^2\beta k/2}$$

Since $k_1 \geq k'$, we have

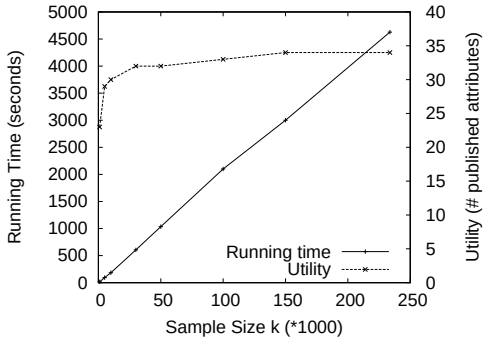
$$Pr[k_1 \leq \alpha\beta k] \leq Pr[k' \leq \alpha\beta k] \leq e^{-(1-\alpha)^2\beta k/2}$$

Hence with probability at least $1 - e^{-(1-\alpha)^2\beta k/2}$, the quasi-identifier has distinct ratio at least $\alpha\beta$ in T_1.

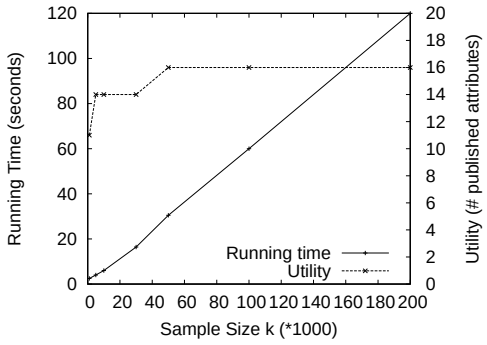
THEOREM 3.2. *With probability at least $1-\delta$, the attribute set published by the algorithm has distinct ratio at most β.*

4 Experiments

We have implemented all algorithms for finding and masking quasi-identifiers, and conducted extensive experiments using real data sets. All experiments were run on a 2.4GHz Pentium PC with 1GB memory.



(a) Masking 0.5-distinct quasi-identifiers



(b) Masking 0.8-separation quasi-identifiers

Figure 1. Performance of masking quasi-identifier algorithms with different sample sizes on table *california*. Figures (a) and (b) show how the running time (the left y axis) and the utility (the right y axis) change with the sample size (the parameter k) in Greedy Approximate algorithms for masking 0.5-distinct and 0.8-separation quasi-identifiers.

4.1 Data Sets

One source of data sets is the census microdata "Public-Use Microdata Samples (PUMS)" [1], provided by US Census Bureau. We gather the 5 percent samples of Census 2000 data from all states and put into a table "census". To study the performance of our algorithms on tables with different sizes, we also extract 1 percent samples of state-level data and select 4 states with different population sizes – Idaho, Washington, Texas and California. We extract 41 attributes including age, sex, race, education level, salary etc. We only use adult records (age $\geq$ 20) because many children are indistinguishable even with all 41 attributes. The table *census* has 10 million distinct adults, and the sizes of *Idaho*, *Washington*, *Texas* and *California* are 8867, 41784, 141130 and 233687 respectively.

We also use two data sets *adult* and *covtype* provided by UCI Machine Learning Repository [21]. The *covtype* table has 581012 rows and 54 attributes. We use 14 attributes of *adult* including age, education level, marital status; the number of records in *adult* is around 30000.

4.2 Masking Quasi-identifiers

The greedy approximate algorithms for masking quasi-identifiers are randomized algorithms that guarantee to satisfy the privacy constraints with probability $1 - \delta$. We set $\delta = 0.01$, and the privacy constraint are satisfied in all experiments, which confirms the accuracy of our algorithms.

Figure 1 shows the tradeoff between the running time and the utility (the number of attributes published), using the *california* data set. Both the running time and the utility decrease as the sample size k decreases; however, the running time decreases linearly with k while the utility degrades very slowly. For example, running the greedy algorithm for masking 0.5-distinct quasi-identifiers on the entire table (without random sampling) takes 80 minutes and publishes 34 attributes (the rightmost point in Figure a); using a sample of 30000 tuples the greedy algorithm takes only 10 minutes and outputs 32 attributes. Figure b shows the impact of k on the masking separation quasi-identifier algorithm. To run the greedy algorithm for masking 0.8-separation quasi-identifier on the entire table takes 728 seconds (not shown in the figure); using a sample of 50000 pairs offers the same utility and only takes 30 seconds. The results show that our random sampling technique can greatly improve time and space complexity (space is also linear in k), with only minor sacrifice on the utility.

Data Sets	Greedy		Greedy Approximate	
	time	utility	time	utility
adult	36s	12	-	-
covtype	-	-	2000s	46
idaho	172s	33	-	-
wa	880s	34	620s	33
texas	3017s	35	630s	33
ca	4628s	34	606s	32
census	-	-	755s	30

Table 2. Algorithms for masking 0.5-distinct quasi-identifiers. The column "Greedy" represents the greedy algorithm on the entire table; the column "Greedy Approximate" represents running greedy algorithm on a random sample of 30000 tuples. We compare the running time and the utility (the number of published attributes) of the two algorithms on different data sets. The results of Greedy on *census* and *covtype* are not available because the algorithm does not terminate in 10 hours; the results of Greedy Approximate on *adult* and *Idaho* are not available because the input tuple number is less than 30000.

Data Sets	Greedy		Greedy Approximate	
	time	utility	time	utility
adult	19s	5	2s	5
covtype	2 hours	38	104s	37
idaho	147s	24	30s	23
wa	646s	23	35s	23
texas	1149s	19	34s	19
ca	728s	16	30s	16
census	-	-	170s	17

Table 3. Algorithms for masking 0.8-separation quasi-identifiers. The column "Greedy" represents the greedy algorithm on the entire table, and the column "Greedy Approximate" represents running greedy algorithm on a random sample of 50000 pairs of tuples. We compare the running time and the utility of the two algorithms on different data sets. The result of Greedy on *census* is unavailable because the algorithm does not terminate in 10 hours.

Table 2 and 3 compare the running time and the utility (the number of published attributes) of running the greedy algorithm on the entire table versus on a random sample (we use a sample of 30000 tuples in Table 2 and a sample of 50000 pairs of tuples in Table 3). Results on all data sets confirm that the random sampling technique is able to reduce the running time dramatically especially for large tables, with only minor impact on the utility. For the largest data set *census*, running the greedy algorithm on the entire table does not terminate in 10 hours, while with random sampling it only takes no more than 13 minutes for masking 0.5-distinct quasi-identifier and 3 minutes for masking 0.8-separation quasi-identifier.

4.3 Approximate Minimum Key Algorithms

Finally we examine the greedy algorithms for finding minimum key and (ϵ, δ)-separation or -distinct minimum key in Section 2. Table 4 shows the experimental results of the Greedy Minimum Key, Greedy $(0.1, 0.01)$-Distinct Minimum Key, and Greedy $(0.001, 0.01)$-Separation Minimum Key algorithms on different data sets.

The Greedy Minimum Key algorithm (applying greedy algorithm directly on the entire table) works well for small data sets such as *adult*, *idaho*, but becomes unaffordable as the data size increases. The approximate algorithms for separation or distinct minimum key are much faster. For the table *California*, the greedy minimum key algorithm takes almost one hour, while the greedy distinct algorithm takes 2.5 minutes, and greedy separation algorithm merely seconds; for the largest table *census*, the greedy minimum key algorithm takes more than 10 hours, while the approximate algorithms take no more than 15 minutes. The space and time requirements of our approximate minimum key algorithms are sublinear in the number of input tuples, and we expect the algorithms to scale well on even larger data sets.

We measure the distinct and separation ratios of the output quasi-identifiers, and find the ratios always within error ϵ. This confirms the accuracy of our algorithms.

Theorem 2.3 and 2.4 provide the theoretical bounds on the size of the quasi-identifiers found by our algorithms ($\ln m$ or $\ln mn$ times the minimum key size). Those bounds are worst case bounds, and in practice we usually get much smaller quasi-identifiers. For example, we find that the minimum key size of *adult* is 13 by exhaustive search, and the greedy algorithm for both distinct and separation minimum key find quasi-identifiers no larger than the minimum key. (For other data sets in Table 4, computing the minimum key exactly takes prohibitively long time, so we are not able to verify the approximation ratio of our algorithms.) We also generate synthetic tables with known minimum key sizes, then apply the greedy distinct minimum key algorithm (with $\epsilon = 0.1$) on those tables and are always able to find quasi-identifiers no larger than the minimum key size. Those experiments show that in practice our approximate minimum key algorithms usually perform much better than the theoretical worst case bounds, and are often able to find quasi-identifiers with high separation (distinct) ratio and size close to the minimum.

5 Related Work

The implication of quasi-identifiers to privacy is first formally studied by Sweeney, who also proposed the k-anonymity framework as a solution to this problem [25, 24]. Afterwards there is numerous work which studies the complexity of this problem [17, 2], designs and implements algorithms to achieve k-anonymity [23, 4], or extends upon the framework [16, 14]. Our algorithm for masking quasi-identifiers can be viewed as an approximation to k-anonymity where the suppression must be conducted at the attribute level. Also it is an "on average" k-anonymity because it does not provide perfect anonymity for every individual but does so for the majority; a similar idea is used in [15]. On the other side, our algorithms for finding keys/quasi-identifiers attempt to attack the privacy of published data from the adversary's point of view, when the publish data is not k-anonymized. To the best of our knowledge, there is no existing work addressing this problem.

Our algorithms exploit the idea of using random samples to trade off between accuracy and space complexity, and can be viewed as streaming algorithms. Streaming algorithms emerged as a hot research topic in the last decade; see [20] for a survey of this area.

Keys are special cases of functional dependencies, and quasi-identifiers are a special case of approximate functional dependency. Our definitions of separation and dis-

Data Sets	Greedy		distinct Greedy ($\epsilon = 0.1$)			separation Greedy ($\epsilon = 0.001$)		
	time	key size	time	key size	distinct ratio	time	key size	separation ratio
adult	35.5s	13	8.8s	13	1.0	3.11s	5	0.99995
covtype	964s	5	78.1s	3	0.9997	27.1s	2	0.999996
idaho	50.4s	14	15.2s	8	0.997	1.07s	3	0.9999
wa	490s	22	34.1s	8	0.995	7.14s	3	0.99993
texas	2032s	29	120s	14	0.995	13.2s	4	0.99995
ca	3307s	29	145s	13	0.994	16.3s	4	0.99998
census	-	-	808s	17	0.993	120s	3	0.99998

Table 4. Running time and output key sizes of the Greedy Minimum Key, Greedy $(0.1, 0.01)$-Distinct Minimum Key, and Greedy $(0.001, 0.01)$-Separation Minimum Key algorithms. The result of Greedy Minimum Key on *census* is not available because the algorithm does not terminate in 10 hours.

tinct ratios for quasi-identifiers are adapted from the measures for quantifying approximations of functional dependencies proposed in [13, 22].

6 Conclusions and Future Work

In this paper, we designed efficient algorithms for discovering and masking quasi-identifiers in large tables. We developed efficient algorithms that find small quasi-identifiers with provable size and separation/distinct ratio guarantees, with space and time complexity sublinear in the number of input tuples. We also designed efficient algorithms for masking quasi-identifiers in large tables.

All algorithms in the paper can be extended to the weighted case, where each attribute is associated with a weight and the size/utility of a set of attributes is defined as the sum of their weights. The idea of using random samples to trade off between accuracy and space complexity can potentially be explored in other problems on large tables.

References

[1] Public-use microdata samples (pums). *http://www.census.gov/main/www/pums.html*.

[2] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D.Thomas, and A. Zhu. Anonymizing tables. In *ICDT*, 2005.

[3] R. Ananthakrishna, S. Chaudhuri, and V. Ganti. Eliminating fuzzy duplicates in data warehouses. In *VLDB*, 2002.

[4] R. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. In *ICDE*, 2005.

[5] M. Charikar, S. Chaudhuri, R. Motwani, and V. Narasayya. Towards estimation error guarantees for distinct values. In *PODS*, 2000.

[6] S. Chaudhuri, K. Ganjam, V. Ganti, and R. Motwani. Robust and efficient fuzzy match for online data cleaning. In *SIGMOD*, 2003.

[7] M. R. Garey and D. S. Johnson. Computers and intractability. 1979.

[8] C. Giannella and E. Robertson. On approximation measures for functional dependencies. *Information Systems*, 2004.

[9] C. M. Giannella, M. M. Dalkilic, D. P. Groth, and E. L. Robertson. Using horizontal-vertical decompositions to improve query evaluation. *LNCS 2405*.

[10] B. Halldorsson, M. Halldorsson, and R. Ravi. Approximability of the minimum test collection problem. In *ESA*, 2001.

[11] Y. Huhtala, J. Karkkainen, P. Porkka, and H. Toivonen. Discovery of functional and approximate dependencies using partitions. In *ICDE*, 1998.

[12] D. Johnson. Approximation algorithms for combinatorial problems. In *J. Comput. System Sci.*, 1974.

[13] J. Kivinen and H. Mannila. Approximate dependency inference from relations. In *Theoretical Computer Science*, 1995.

[14] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *ICDE*, 2007.

[15] S. Lodha and D. Thomas. Probabilistic anonymity. *Technical Report*.

[16] Machanavajjhala, J. Gehrke, and D. Kifer. l-diversity: privacy beyond k-anonymity. In *ICDE*, 2006.

[17] A. Meyerson and R. Williams. On the complexity of optimal k-anonymity. In *PODS*, 2004.

[18] B. Moret and H. Shapiro. On minimizing a set of tests. In *SIAM Journal on Scientific and Statistical Computing*, 1985.

[19] R. Motwani and P. Raghavan. Randomized algorithm. 1995.

[20] S. Muthukrishnan. Data streams: Algorithms and applications. 2005.

[21] D. Newman, S. Hettich, C. Blake, and C. Merz. Uci repository of machine learning databases. *http://www.ics.uci.edu/∼mlearn/MLRepository.html*.

[22] B. Pfahringer and S. Kramer. Compression-based evaluation of partial determinations. In *SIGKDD*, 1995.

[23] P. Samarati and L. Sweeney. Generalizing data to provide anonymity when disclosing information. In *PODS*, 1998.

[24] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. In *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 2002.

[25] L. Sweeney. k-anonymity: a model for protecting privacy. In *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 2002.

On the Lindell-Pinkas Secure Computation of Logarithms: From Theory to Practice

Raphael S. Ryger*
Yale University
New Haven, CT USA
ryger@cs.yale.edu

Onur Kardes†
Stevens Institute of Technology
Hoboken, NJ USA
onur@cs.stevens.edu

Rebecca N. Wright†
Rutgers University
Piscataway, NJ USA
rebecca.wright@rutgers.edu

Abstract

Lindell and Pinkas demonstrated that it is feasible to preserve privacy in data mining by employing a combination of general-purpose and specialized secure-multiparty-computation (SMC) protocol components. Yet practical obstacles of several sorts have impeded a fully practical realization of this idea. In this paper, we address the correctness and practicality of one of their primary contributions, a secure natural logarithm computation, which is a building block crucial to an SMC approach to privacy-preserving data mining applications including construction of ID3 trees and Bayesian networks. We first demonstrate a minor error in the Lindell-Pinkas solution, then provide a correction along with several optimizations. We explore a modest trade-off of perfect secrecy for a performance advantage, a strategy that adds flexibility in the effective application of hybrid SMC to data mining.

1 Introduction

Privacy-preservation objectives in data mining can often be framed ideally as instances of secure multiparty computation (SMC), wherein multiple parties cooperate in a computation without thereby learning each other's inputs. The characterization of SMC is very encompassing, admitting a great variety of input and output configurations, so that a general recipe for adding the SMC input security to arbitrary well-specified multiparty computations would seem to solve many quite different problems in one fell swoop. Indeed, general approaches to SMC were proposed for a variety of settings already in the 1980s. Yet the framing of privacy preservation for particular data-mining tasks as SMC problems, making them amenable to the general approaches, is usually not useful. For all but the most trivial computations, the general SMC solutions have been too cumbersome to apply and would be impractical to run. They require the computation to be represented as an algebraic circuit, with all loops unrolled to as many iterations as would possibly be needed for the supported inputs, and with all contingent branches of the logic as conventionally expressed—such as iterations that happen not to be needed—executed in every run regardless of the inputs. One may reasonably conclude that SMC is just a theoretical curiosity, not relevant for real-world privacy-preserving data mining, where inputs are not just a few bits but rather entire databases.

Lindell and Pinkas [LP00, LP02] have shown the latter conclusion to be inappropriate. A privacy-preserving data-mining task, they point out, need not be cast as a monolithic SMC problem to which to apply an expensive general SMC solution. Instead, the task may be decomposed into modules requiring SMC, all within a computational superstructure that may itself admissibly be left public. The modules requiring SMC may, in part, be implemented with special-purpose protocols with good performance, leaving general SMC as a fallback (at the module-implementation level) only where special approaches have not been found. The key to such construction is that we are able to ensure secure chaining of the secure protocol components. We prevent information from leaking at the seams between the SMC components by having them produce not public intermediate outputs but rather individual-party shares of the logical outputs, shares that may then be fed as inputs to further SMC components. Lindell and Pinkas illustrate this creative, hybrid methodology by designing a two-party SMC version of the ID3 data-mining algorithm for building a classification tree, a query-sequencing strategy for predicting an unknown attribute—e.g., loan worthiness—of a new entity whose other attributes—e.g., those characterizing credit history, assets, and income—are obtainable by (cost-bearing) query. At each construction step, the

*Supported in part by ONR grant N00014-01-1-0795 and by US-Israel BSF grant 2002065.
†Supported in part by NSF grant 0331584.

P3DM'08, April 26, 2008, Atlanta, Georgia, USA.

ID3 algorithm enters an episode of information-theoretic analysis of the database of known-entity attributes. The privacy concern is introduced, in the Lindell-Pinkas setting, by horizontal partitioning of that database between two parties that must not share their records. The computation is to go through as if the parties have pooled their data, yet without them revealing to each other in their computational cooperation any more regarding their private data than is implied by the ultimate result that is to be made known to them both.

While demonstrating the potential in a modular SMC approach to prospective appliers of the theory, Lindell and Pinkas offer SMC researchers and implementors some design suggestions for particular SMC modules needed in their structuring of the two-party ID3 computation. Strikingly, they need only three such SMC modules, all relatively small and clearly useful for building other protocols, namely, shares-to-shares logarithm and product protocols and a shares-to-public-value minindex protocol. Their intriguing recommendation for the secure logarithm protocol, critical to the accuracy and performance of SMC data mining whenever information-theoretic analysis is involved, is our focus in this paper.

The present authors have been engaged in a privacy-preserving data-mining project [YW06, KRWF05] very much inspired by Lindell and Pinkas. Our setting is similar: a database is arbitrarily partitioned between two parties wishing to keep their portions of the data private to the extent that is consistent with achieving their shared objective of discovering a Bayes-net structure in their combined data. The information-theoretic considerations and the scoring formula they lead to are very similar to those in the ID3 algorithm for classification-strategy building, as is the external flow of control that invokes scoring on candidate next query attributes given a set of query attributes that has already been decided upon. (The details and their differences are not germane to the present discussion.) The adaptation we do for privacy preservation in our two-party setting is, not surprisingly, very similar to what Lindell and Pinkas do. Indeed, we need the same SMC components that they do and just one more, for computing scalar products of binary-valued vectors. The latter additional need has more to do with the difference in setting—we are admitting arbitrary, rather than just horizontal, partitioning of the data—than with the difference in analytical objective. In fact, our software would not require much adjustment to serve as a privacy-preserving two-party ID3 implementation—in fact, supporting arbitrarily partitioned data, given the incorporated scalar-product component.

Launching our investigation a few years after Lindell and Pinkas's paper, we have had the advantage of the availability of the Fairplay system [MNPS04] for actually implementing the Yao-protocol components. We have created tools to support the entire methodology, enabling us to take our protocol from a theoretical suggestion all the way to usable software. This exercise has been illuminating. On one hand, it has produced the most convincing vindication of which we are aware of Lindell and Pinkas' broad thesis regarding the practical achievability of SMC in data mining while teaching us much about the software engineering required for complex SMC protocols. On the other hand, as is typical in implementation work, it has revealed flaws in a number of areas of the underlying theoretical work, including our own. In this paper, we present our observations on the Lindell-Pinkas logarithm proposal. We correct a mathematical oversight and address a general modular-SMC issue that it highlights, the disposition of scaling factors that creep into intermediate results for technical reasons.

We begin in Section 2 with a careful account of the Lindell-Pinkas proposal for a precision-configurable secure two-party shares-to-shares computation of natural logarithms. In Section 3, we explain the mathematical oversight in the original proposal and show that the cost of a straightforward fix by scale-up is surprisingly low, although leaving us with a greatly inflated scale-up factor. In Sections 4 and 5, we propose efficient alternatives for doing arbitrary scaling securely. These enable a significant optimization in the first phase of the Lindell-Pinkas protocol, allowing removal of the table look-up from the Yao circuit evaluation. We briefly point out the effectiveness of a simple dodge of most of the problematics of the Lindell-Pinkas protocol in Section 6. We conclude with a discussion of our implementation of the revised Lindell-Pinkas protocol and its performance in Section 7.

2 The Lindell-Pinkas $\ln x$ protocol

The Lindell-Pinkas proposed protocol for securely computing $\ln x$ is intended as a component in a larger secure two-party protocol. The parties are presumed not to know, and must not hereby learn, either the argument or its logarithm. They contribute secret shares of the argument and obtain secret shares of its logarithm. The proposed design for this protocol module is itself modular, proceeding in two chained phases involving different technology. The first phase internally determines n and ε such that $x = 2^n(1+\varepsilon)$ with $-1/4 \leq \varepsilon < 1/2$. Note that, since n is an approximate base-2 logarithm of x, the first phase gets us most of the way to the desired logarithm of x. Furthermore, this phase dominates the performance time of the en-

tire logarithm protocol: in absence of a specialized SMC protocol for the first phase, Lindell and Pinkas fall back to dictating it be implemented using Yao's general approach to secure two-party computation, entailing gate-by-gate cryptography-laden evaluation of an obfuscated Boolean circuit. Yet the main thrust of the Lindell-Pinkas recommendation is in the second phase, which takes (the secret shares of) ε delivered by phase one and computes an additive correction to the logarithm approximation delivered (as secret shares) by phase one.

We will return to the performance-critical considerations in implementing phase one, not addressed by Lindell and Pinkas. We assume that its Boolean circuitry reconstitutes x from its shares; consults the top 1-bit in its binary representation and the value of the bit following it to determine n and ε as defined; represents n and ε in a manner to be discussed; and returns shares of these representations to the respective parties. These values allow an additive decomposition of the sought natural logarithm of x,

$$\ln x \;=\; \ln\, 2^n(1+\varepsilon) \;=\; n\ln 2 + \ln(1+\varepsilon) \tag{2.1}$$

The purpose is to take advantage of the Taylor expansion of the latter term,

$$\ln(1+\varepsilon) \;=\; \sum_{i=1}^{\infty} \frac{(-1)^{i-1}\varepsilon^i}{i} \;=\; \varepsilon - \frac{\varepsilon^2}{2} + \frac{\varepsilon^3}{3} - \frac{\varepsilon^4}{4} + \cdots \tag{2.2}$$

to enable, in phase two, correction of the phase-one approximation of the logarithm with configurable precision by choice of the number of series terms to be used—a parameter k to be agreed upon by the parties. The computation in the second, refining phase is to proceed by oblivious polynomial evaluation, a specialized SMC technology which is inexpensive compared to the Yao protocol of the first phase.

In this rough mathematical plan, the value ε to be passed from phase one to phase two is a (generally non-integer) rational and the terms in the decomposition of the final result in equation (2.1) are (generally non-integer) reals, whereas the values we will accept and produce in the two SMC phases are most naturally viewed as integers. We are, then, representing the rational and the reals as integers through scale-up and finite-precision approximation. We have considerable latitude in choice of the scale-up factors, particularly considering that the scale-up of a logarithm is just the logarithm to a different base—just as good for information-theoretic purposes as long as the base is used consistently. Still, several considerations inform our choice of scale-up factors. We want the scale-ups to preserve enough precision. On the other hand, there is a performance penalty, here and elsewhere in the larger computation to which this component is contributing, especially in Yao-protocol episodes, for processing additional bits. The chosen scale-up must work mathematically within the larger computation. If an adjustment of the scaling were to be needed for compatibility with the rest of the computation—other than further scale-up by an integer factor—it would entail another secure computation. (We return to this issue in §4.) For the Lindell-Pinkas ID3 computation or for our Bayes-net structure-discovery computation, both information-theoretic, no adjustment would be needed. All the terms added and subtracted to get scores within the larger computation would be scaled similarly, and those scaled scores serve only in comparison with each other.

We assume that the parties have common knowledge of some upper bound N on n, the approximate base-2 logarithm of the input x, and we have phase one deliver the rational ε scaled up by 2^N. This loses no information, deferring control of the precision of the correction term, $\ln\, 2^n(1+\varepsilon)$ in some scale-up, to phase two. Bearing in mind that the slope of the natural-logarithm function is around 1 in the interval around 1 to which we are constraining $1+\varepsilon$, we aim for a scale-up of the correction term by at least 2^N, and plan to scale up the main term of the decomposition, $n\ln 2$, to match. Lindell and Pinkas suggest that the mapping from n to $n\ln 2\cdot 2^N$ be done by table look-up within the Yao protocol of phase one. Any further integer scale-up of the main term to match the scaling of the correction term can be done autonomously by the parties, without SMC, by modular multiplication of their respective shares.

Lindell and Pinkas stipulate that the sharing be with respect to a finite field $\mathcal{F}$ that is large enough in a sense we discuss in more detail in Section 3. A non-field ring will do provided that any particular needed inverses exist. This allows us, e.g., to use Paillier homomorphic encryption in a $\mathbb{Z}_{pq}$ both for the oblivious polynomial evaluation needed in phase two of this logarithm component and, subsequently in the larger computation, for the shares-to-shares secure multiplication to compute $x\ln x$—without additional secure Yao computations to convert the sharing from one modulus to another. The only inverses Lindell and Pinkas need here are of powers of 2, and these would be available in $\mathbb{Z}_{pq}$.

The set-up for phase two, then, preserving the Lindell-Pinkas notation, is that phase one has delivered to the parties, respectively, shares β_1 and β_2 such that $\beta_1 +_{\mathcal{F}} \beta_2 = n\ln 2\cdot 2^N$, toward (whatever ultimate scale-up of) the main term of the decomposition (2.1); and shares α_1 and α_2 such that $\alpha_1 +_{\mathcal{F}} \alpha_2 = \varepsilon\cdot 2^N$, toward the phase-two computation of (the scale-up of) the correction term of the decomposition. We continue to

phase two.

Replacing ε in formula (2.2) with $(\alpha_1 +_{\mathcal{F}} \alpha_2)/2^N$, we get

$$\ln(1+\varepsilon) \;=\; \sum_{i=1}^{\infty} \frac{(-1)^{i-1}(\alpha_1 +_{\mathcal{F}} \alpha_2)^i}{i\ 2^{Ni}} \tag{2.3}$$

In this infinite-series expression, the only operation to be carried out in the finite ring $\mathcal{F}$ is the recombination of the shares, α_1 and α_2, as noted. The objective in phase two is to compute the series in sufficiently good approximation through oblivious polynomial evaluation by the two parties, returning shares of the value to the parties. So we need to get from the infinite series—a specification of a limit in $\mathbb{R}$ for what appear to be operations in $\mathbb{Q}$—to a polynomial over the finite ring $\mathcal{F}$ that may be evaluated so as to contribute to the sought shares. This will entail several steps of transformation.

Step 1. The computation must be finite. We take only k terms of the series.

Step 2. We deal somehow with the division that appears in the summand. We need to be sure we end up, when the transformation is complete, with a polynomial over $\mathcal{F}$. We can scale up the whole formula to cancel some or all of the division. The disposition of any remaining division, as we work toward determining the coefficients of the polynomial to be evaluated, turns out to be problematic, largely occasioning this paper. (The existence of modular inverses in $\mathcal{F}$ for the remaining divisors is not sufficient.) For the moment, let σ be whatever scale-up factor we decide to use here.

Step 3. We *reinterpret* the outer summation and the multiplication, including the binomial exponentiation and the multiplication by σ, as modular addition and multiplications in $\mathcal{F}$. Note that we cannot even open the parentheses by formal exponentiation, applying a distributive law, without first reinterpreting the multiplication as in $\mathcal{F}$. We have no law regarding the distribution of multiplication in $\mathbb{Z}$ over addition in $\mathcal{F}$. This requires that we assure ourselves that the reinterpretation does not alter the value of the expression. Lindell and Pinkas ensure this by requiring $\mathcal{F}$ to be sufficiently large, and we will review the consideration.

Step 4. We replace the occurrence of 'α_2' in (2.3)—as truncated, division-resolved, and modularly reinterpreted—with the variable 'y'. Knowing α_1, party 1 does the formal exponentiations and collects terms, all modulo $|\mathcal{F}|$, yielding a polynomial in 'y' over $\mathcal{F}$. Party 1 randomly chooses $z_1 \in \mathcal{F}$ and subtracts it from the constant term of the polynomial. Where $Q(y)$ is the resulting polynomial and z_2 is its value at $y = \alpha_2$, to be obtained by party 2 through the oblivious polynomial evaluation to follow, we have

$$z_2 \;=\; Q(y)\,|_{y=\alpha_2} \;=\; \sum_{i=1}^{k} \frac{\sigma(-1)^{i-1}(\alpha_1 + y)^i}{i\ 2^{Ni}} - z_1 \Bigg|_{y=\alpha_2} \tag{2.4}$$

where all operations—once the approach to the division in the summand is sorted out—are in $\mathcal{F}$, so that

$$z_1 +_{\mathcal{F}} z_2 \;\approx\; \sum_{i=1}^{\infty} \frac{\sigma(-1)^{i-1}(\alpha_1 +_{\mathcal{F}} \alpha_2)^i}{i\ 2^{Ni}} \;=\; \ln(1+\varepsilon)\cdot\sigma$$

—all operations here, except as indicated, back in $\mathbb{R}$. Thus, the computation of z_2 according to (2.4) by oblivious polynomial evaluation accomplishes the sharing of $\ln(1+\varepsilon)\cdot\sigma$ as z_1 and z_2. The parties may autonomously modularly multiply β_1 and β_2 by $\mathrm{lcm}(2^N, \sigma)/2^N$, giving β_1' and β_2', respectively; and modularly multiply z_1 and z_2 by by $\mathrm{lcm}(2^N, \sigma)/\sigma$, giving z_1' and z_2', respectively; and modularly add their respective results from these scale-ups. Then, per the decomposition in (2.1),

$$\begin{aligned}(\beta_1' +_{\mathcal{F}} z_1') +_{\mathcal{F}} (\beta_2' +_{\mathcal{F}} z_2') \;&=\; (\beta_1' +_{\mathcal{F}} \beta_2') +_{\mathcal{F}} (z_1' +_{\mathcal{F}} z_2') \\ \approx\; (n\ln 2 + \ln(1+\varepsilon))\cdot \mathrm{lcm}(2^N, \sigma) \;&=\; \ln x \cdot \mathrm{lcm}(2^N, \sigma)\end{aligned}$$

accomplishing the original goal of securely computing shares of $\ln x$ from shares of x—if with a scale-up that we hope is innocuous. But this sketch of the protocol still needs to be fleshed out. We back up now, first briefly to step 3, and then to step 2, our main focus.

By the time we get to step 3, we should be left with an expression prescribing finitely many operations in $\mathbb{Z}$, viewing $+_{\mathcal{F}}$ as an operation in $\mathbb{Z}$ and viewing division as a partially-defined operation in $\mathbb{Z}$. Looking ahead to step 4, we will be replacing the occurrences of 'α_2' in this expression with the variable 'y' and algebraically reorganizing it into the polynomial $Q(y)$ (with a change to the constant term). In this step 3, we change only the semantics of the expression arrived at, not its syntactic composition. The claim to be made is that the hybrid expression at hand, involving some modular additions but otherwise non-modular operations, can be reinterpreted to involve only modular operations without change to the induced expression value—allowing the expression then to be transformed syntactically with guarantee of preservation of value, but now with respect to the new semantics. This tricky claim, made implicitly, bears explicit examination. We can frame the issue abstractly. Suppose φ is an arbitrarily complex numerical expression built recursively of variables and function symbols (admitting constants as 0-ary function symbols). We have a conventional interpretation of φ in the domain $\mathbb{Z}$. We also have an alternate interpretation

of φ in the domain $\mathbb{Z}_m$. Furthermore, we have an alternate expression, φ', obtained from φ by transformations guaranteed to preserve the value of the whole under the interpretation in $\mathbb{Z}_m$ for any assignment of values from $\mathbb{Z}_m$ to the variables. We intend to compute φ' as interpreted in $\mathbb{Z}_m$. Under what circumstances can we be assured that this computation will yield the same value as does evaluation of the original expression φ according to the original interpretation in $\mathbb{Z}$? In the case at hand, φ is

$$\sum_{i=1}^{k} \frac{\sigma(-1)^{i-1}(\alpha_1 +_{\mathcal{F}} y)^i}{i\ 2^{Ni}} \tag{2.5}$$

(with some decision as to how to interpret the division), whereas φ' is

$$Q(y) + z_1$$

to be interpreted in $\mathbb{Z}_m$ (where $m = |\mathcal{F}|$) and be so computed, with the value to be assigned to 'y' in both cases being α_2.

There are obvious strong sufficient conditions under which modular reinterpretation preserves value. We do have to be careful to take into account, in generalizing, that in our instance ε may be negative, and that our summation expression has sign alternation, so we need to proceed via a "signed-modular" interpretation, wherein the mod-m integers $\lceil \frac{m}{2} \rceil$ to $m-1$ are viewed as "negative", i.e., they are isomorphically replaced by the integers $-\lfloor \frac{m}{2} \rfloor$ to -1. (Choosing the midpoint for the cutover here is arbitrary, in principle, but appropriate for our instance.) If (a) for the values we are contemplating assigning to the variables, the recursive evaluation of φ under the original interpretation assigns values to the subexpressions of φ that are always integers in the interval $[-\lfloor \frac{m}{2} \rfloor, \lfloor \frac{m}{2} \rfloor]$; and if (b) the functions assigned to the function symbols in the signed-modular reinterpretation agree with the functions assigned by the original interpretation whenever the arguments and their image under the original function are all in that signed-mod-m interval; then the signed-modular reinterpretation will agree with the original interpretation on the whole expression φ for the contemplated value assignments to the variables. Note that we need not assume that the reinterpretation associates with the function symbols the signed-modular analogues of the original functions, although this would ensure (b). Nor would a stipulation of modular agreement be sufficient, in general, without condition (a), even if the original evaluation produces only (overall) values in the signed-mod-m domain for value assignments of interest. The danger is that modular reduction of intermediate values, if needed, may lose information present in the original evaluation. In our case, the single variable, 'y', is assigned the value α_2, which may be as large as, but no larger than, $m-1$. The constant α_1 is similarly less than m. We can view these mod-m values, returned by the Yao protocol in phase one, as being the corresponding signed-mod-m values instead, with $+_{\mathcal{F}}$ operating on them isomorphically. Moreover, $\alpha_1 +_{\mathcal{F}} y$ then evaluates into the interval $[-\frac{1}{4}2^N, \frac{1}{2}2^N)$, where we can arrange for the endpoints to be *much* smaller in absolute value than $\lfloor \frac{m}{2} \rfloor$. This allows Lindell and Pinkas to reason about setting m high enough so that indeed all subexpressions of our φ will evaluate, in the original interpretation, into the signed-mod-m domain. Note that if formal powers of 'α_1' and of 'y' appeared as subexpressions in our original expression φ, as they do in our φ', the polynomial $Q(y)+z_1$ which we actually compute, we would have concern over potential loss of information in modular reduction impeding the modular reinterpretation; but the power subexpressions appear only *after* we have reinterpreted and transformed φ, and are by then of no concern.

We now return to step 2, attending to the division in the Taylor-series terms.

3 The division problem

We have already seen that choices of scaling factor are governed by several considerations including preservation of precision, avoidance of division where it cannot be carried out exactly, and compatibility among intermediate results. For preservation of precision, we have been aiming to compute the main and correction terms of (2.1) scaled up by at least 2^N. Lindell and Pinkas incorporate this factor into their σ in preparing the polynomial. To dispose of the i factors in the denominator in (2.4), they increase the scale-up by a factor of $\mathrm{lcm}(2,\ldots,k)$. With σ now at $2^N\mathrm{lcm}(2,\ldots,k)$, the truncated Taylor series we are looking at in step 2 becomes

$$\ln(1+\varepsilon)\cdot 2^N\mathrm{lcm}(2,\ldots,k)\ \approx \sum_{i=1}^{k} \frac{(-1)^{i-1}\ (\mathrm{lcm}(2,\ldots,k)/i)\ (\alpha_1 +_{\mathcal{F}} \alpha_2)^i}{2^{N(i-1)}} \tag{3.6}$$

We know that in step 3 we will be reinterpreting the operations in this expression—more precisely, in the expression we intend this expression to suggest—as operations in $\mathcal{F}$. Clearly, since k is agreed upon before the computation, the subexpression '$\mathrm{lcm}(2,\ldots,k)/i$' may be replaced immediately by (a token for) its integer value. We are still left with a divisor of $2^{N(i-1)}$, but Lindell and Pinkas reason that $(\alpha_1 +_{\mathcal{F}} \alpha_2)^i$, although not determined until run time, will be divisible by $2^{N(i-1)}$. After all, $(\alpha_1 +_{\mathcal{F}} \alpha_2)^i$ will be $(\varepsilon \cdot 2^N)^i$, and the denominator was designed expressly to divide this

to leave $\varepsilon^i \cdot 2^N$. Apparently, all we need to do is allow the division bar to be reinterpreted in step 3 as the (partially defined) division operation in $\mathbb{Z}_m$, i.e., multiplication by the modular inverse of the divisor. We can assume that m is not even, so that powers of 2 have inverses modulo m. Furthermore, whenever a divides b (in $\mathbb{Z}$) and $b < m$, if a has an inverse ($a \in \mathbb{Z}_m^*$) then $a^{-1}b$ in $\mathbb{Z}_m$ is just the integer b/a. It would appear that the strong sufficient conditions for reinterpretation are met.

The trouble is that, although $(\alpha_1 +_{\mathcal{F}} \alpha_2)^i = (\varepsilon \cdot 2^N)^i$ is an integer smaller than m (given that we will ensure that m is large enough) and although the *expression* '$(\varepsilon \cdot 2^N)^i$' appears to be formally divisible by the expression '$2^{N(i-1)}$', the integer $(\varepsilon \cdot 2^N)^i$ is not, in general, divisible by the integer $2^{N(i-1)}$. In $\mathbb{Q}$, the division indeed yields $\varepsilon^i 2^N$, which is just the scale-up by 2^N we engineered it to achieve. That rational scale-up is an integer for $i = 1$, but will generally not be an integer for $i > 1$. (Roughly, $\varepsilon^i 2^N$ is an integer if the lowest-order 1 bit in the binary representation of x is within N/i digits of its highest-order 1 bit—a condition that excludes most values of x already for $i = 2$.) This undermines the sufficient condition Lindell and Pinkas hoped to rely on to justify the modular reinterpretation, our step 3. Without the divisibility in the integers, there is no reason to believe that reinterpretation of the division by $2^{N(i-1)}$ as modular multiplication by its mod-m inverse $(2^{N(i-1)})^{-1}$ would have anything to do with the approximation we thought we were computing. The ensuing formal manipulation in step 4 to get to a polynomial to be evaluated obliviously would be irrelevant.

The immediate brute-force recourse is to increase the scale-up factor, σ, currently at $2^N \mathrm{lcm}(2, \ldots, k)$, to $2^{Nk}\mathrm{lcm}(2, \ldots, k)$. This leaves our truncated Taylor series as

$$\ln(1+\varepsilon) \cdot 2^{Nk}\mathrm{lcm}(2, \ldots, k) \approx$$
$$\sum_{i=1}^{k} (-1)^{i-1}\, 2^{N(k-i)} (\mathrm{lcm}(2, \ldots, k)/i)\, (\alpha_1 +_{\mathcal{F}} \alpha_2)^i \quad (3.7)$$

Phase one still feeds phase two shares of ε scaled up by 2^N. For compatibility with the larger scale-up of the correction term of the decomposition as now delivered (in shares) by phase two, the parties will autonomously scale up their shares of the main term of the decomposition by a further factor of $2^{N(k-1)}$.

The natural concern that a scaling factor so much larger will require $\mathcal{F}$ to be much larger, with adverse performance implications, turns out to be unfounded. Surprisingly, the guideline given by Lindell and Pinkas for the size of $\mathcal{F}$—namely, 2^{Nk+2k} or more—need not be increased by much. The original guideline actually remains sufficient for the step-3 reinterpretation of the operations to be sound. But now, with the (unshared) scaled-up correction term alone so much wider, requiring some 2^{Nk} bits of representation, we are in danger of running out of room in the space for the scaled-up main term if $\log_2 N > 2k$. Raising the size requirement for $\mathcal{F}$ to $2^{Nk+2k+\log_2 N}$ should be sufficient. If we want to provide, in the larger protocol, for computation of $x \ln x$, scaled up to $x(\sigma \ln x)$, in the same space $\mathcal{F}$, we need to raise the size requirement for $\mathcal{F}$ to $2^{Nk+2k+\log_2 N+N}$

Our larger scale-up here does not carry any additional information, of course. The creeping growth in the computational space does affect performance, but only minimally. Even in Yao SMC episodes, the larger space affects only the modular addition to reconstitute shared inputs at the outset and the modular addition to share the computed results at the end. The computation proper is affected by the size of the space of the actual unshared inputs, but not by the size of the space for modular sharing.

The more significant issue is that we continue to be saddled with scaling factors that are best not incurred in building blocks intended for general use. We explore efficient ways to reverse unwanted scaling. The problem is tantamount to that of efficiently *introducing* wanted arbitrary—i.e., not necessarily integral—scaling. Lindell and Pinkas need such scaling to get from base-2 logarithms to natural logarithms in phase one of the protocol. A good solution to this problem of secure arbitrary scaling will enable us to do better than (even a smart implementation of) the table look-up inside the phase-one Yao protocol that they call for, in addition to allowing reversal of whatever scale-up is delivered by the entire logarithm protocol.

4 Secure non-integer scaling of shared values

Suppose parties 1 and 2 hold secret shares modulo m, respectively γ_1 and γ_2, of a value γ; and suppose $\sigma = \kappa + \rho$ is a scaling factor to be applied to γ, where κ is a non-negative integer and $0 \le \rho < 1$. $\sigma\gamma$ is not, in general, an integer, but a solution that can provide the parties shares of an integer approximation of $\sigma\gamma$ suffices. $\kappa\gamma$ may be shared exactly simply by having the parties autonomously modularly scale up their shares by κ. That leaves the sharing of (an approximation of) $\rho\gamma$, the shares to be added modularly to the shares of $\kappa\gamma$ to obtain shares of (an approximation of) $\sigma\gamma$. The problem is that approximate multiplication by a non-integer does not distribute over modular addition, even approximately!

A bifurcated distributive property does hold, however. If the ordinary sum $\gamma_1 + \gamma_2$ is $< m$, the usual distributive law for multiplication of the sum by ρ holds

approximately for approximate multiplication. If, on the other hand, the ordinary sum $\gamma_1 + \gamma_2$ is $\geq m$, then the modular sum is, in ordinary terms, $\gamma_1 + \gamma_2 - m$, so that the distribution of the multiplication by ρ over the modular addition of γ_1 and γ_2 will need an adjustment of approximately $-\rho m$. This suggests the following protocol to accomplish the scaling by ρ mostly by autonomous computation by the parties on their own shares, but with a very minimal recourse to a Yao protocol to select between the two cases just enumerated. The Yao computation takes $\rho\gamma_1$ and $\rho\gamma_2$, each rounded to the nearest integer, as computed by the respective parties; and the original shares γ_1 and γ_2 as well. Party 1 also supplies a secret random input $z_1 < m$. The circuit returns to party 2 either $(\rho\gamma_1 + \rho\gamma_2) +_{\mathrm{mod}\, m} z$ or $(\rho\gamma_1 + \rho\gamma_2 - \rho m) +_{\mathrm{mod}\, m} z$ accordingly as $\gamma_1 + \gamma_2 < m$ or not. Party 1's share is $m - z_1$. The integer approximation of ρm is built into the circuit. The cumulative approximation error is less than 1.5, and usually less than 1.

But an unconventional approach can allow us to do better still.

5 The practical power of imperfect secrecy

In implementing secure protocols, we tend to be induced by different considerations to choose moduli for sharing that are vastly larger than the largest value that will be shared. In the Lindell-Pinkas logarithm proposal, for instance, if N is 13, as to accommodate ID3 database record counts of around 8,000, and k is 4, our share space is of a size greater than 10^{20}. Prior to our correction, logarithms are to be returned scaled up by around 10^5, making for a maximum output of around 10^6. Thus, the size of the sharing space is larger than the largest shared value by a factor of 10^{14}. In such a configuration, it is a bit misleading to state that the distributive law is bifurcated. The case of the shares *not* jointly exceeding the modulus is very improbable. If we could *assume* the nearly certain case of the shares being excessive—i.e., needing modular reduction—to hold, we would not need a Yao episode to select between two versions of the scaling computation. Each party would scale autonomously and party 1 would subtract ρm to correct for the excess.

We could abide the very small chance of error in this assumption. But better would be to guarantee (approximate) correctness of the autonomous scaling by contriving to *ensure* that the shares be excessive. This turns out to be quite tricky in theory while straightforward in practice. It entails a small sacrifice of the information-theoretic perfection of the secrecy in the sharing, but the sacrifice should be of no practical significance.

Let t be the largest value to be shared, much smaller than the modulus m. We can ensure that shares are excessive by restricting the independently set share to be greater than t. But we can show that if it is agreed that the independent share will be chosen uniformly randomly from the interval $[t+1, m-1]$ then, if it is actually chosen within t of either end of this interval, information will leak to the other party through the complementary share given him for certain of the values from $[0, t]$ that might be shared—to the point of completely revealing the value to the other party in the extreme case. If the choice is at least t away from the ends of the choice interval, perfect secrecy is maintained. But if we take this to heart and agree that the independent share must be from the smaller interval $[2t + 1, m - 1 - t]$ then the same argument can be made regarding the possibility that the choice is actually within t of the ends of this smaller interval. Recursively, to preserve secrecy, we would lop off the ends of the choice interval until nothing was left.

But as in the "surprise quiz" (or "unexpected hanging") paradox, wherein we establish that it is impossible to give a surprise quiz "some day next week," the conclusion here, too, is absurd from a practical point of view. If the independent share is chosen from some huge, but *undeclared*, interval around $m/2$, huge by comparison with t but tiny by comparison with m, there simply is no problem with loss of secrecy. We can assume that the sharing is excessive, and arbitrary scaling can be accomplished by the parties completely autonomously.

We may be able to look at the random choice of the independent share from an undeclared interval instead as a non-uniform random choice, the distribution being almost flat, with the peak probability around $m/2$ dropping off extremely gradually to 0 as the ends of $[t + 1, m - 1]$ are approached. As long as the probabilities are essentially the same in a cell of radius t around whatever independent share is actually chosen—and it is exceedingly unlikely that there not exist a complete such cell around the choice—secrecy is preserved. But theorizing about the epistemology here is beyond our scope. The point is that, in practice, it seems worth considering that we can gain performance by not requiring Yao episodes when non-integer scaling is needed.

In the Lindell-Pinkas protocol, for scaling the approximate base-2 logarithms determined in phase one to corresponding approximate natural logarithms, this approach is fine. For getting rid of the scale-up delivered in the final result, beyond whatever scale-up is sufficient for the precision we wish to preserve, we would need to extend the size of $\mathcal{F}$ somewhat before using this

approach, now that our correction has greatly increased the maximum value that may be delivered as shares by the oblivious polynomial evaluation. On balance, considering the added expense that would be incurred in other components of the larger protocol, it is best not to enlarge $\mathcal{F}$ (further) and to reverse the scaling of the result, if necessary, by the method of the preceding section.

6 Alternative: Pretty good precision, high performance

For many purposes, a much simpler secure computation for logarithms may offer adequate precision. The base is often not important, as noted, so base 2 may do—as indeed it would in the ID3 computation. Noting that in the interval $[1, 2]$ the functions $y = \log_2 x$ and $y = x - 1$ agree at the ends of the interval and deviate by only 0.085 in the middle, we have the Yao circuit determine the floor of the base-2 logarithm and then append to its binary representation the four bits of the argument following its top 1-bit. This gives a result within 1/16 of the desired base-2 logarithm. We used this approach in our Bayes-net structure computation [YW06, KRWF05] while sorting out the issues with the much more complex Lindell-Pinkas proposal. As in the Lindell-Pinkas secure ID3 computation, the logarithms inform scores that, in turn, are significant only in how they compare with other scores, not in their absolute values. As long as the sense of these score comparisons is not affected, inaccuracies in the logarithms are tolerable. We bear in mind also that, in the particular data-mining contexts we are addressing, the algorithms are based on taking the database as a predictive sample of a larger space. In so depending on the database, they are subject to what may be regarded as sampling error in any case. From that perspective, even the reversal of sense in some comparisons of close scores cannot be regarded as rendering the approach inappropriate.

However, as much simpler as this approach is, the performance consideration in its favor is considerably weakened once we remove the conversion from base-2 to scaled-up natural logarithms from the Yao portion of the Lindell-Pinkas protocol, as we now see we can do.

7 Implementation and performance

We have evolved an array of tools to aid in developing hybrid-SMC protocols of the style demonstrated by Lindell and Pinkas. These will be documented in a Yale Computer Science Department technical report and will be made available. Among the resources are a library of Perl functions offering a level of abstraction and control we have found useful for specifying the generation of Boolean circuits; scripts for testing circuits without the overhead of secure computation; particular circuit generators, as for the phase-one Yao episode in the Lindell-Pinkas logarithm protocol and for the minindex Yao episode needed for the best-score selection in their larger secure ID3 computation; additional SMC components not involving circuits; and a library of Perl functions facilitating the coordination of an entire hybrid-SMC computation involving two parties across a network.

We have been developing and experimenting on NetBSD and Linux operating systems running on Intel Pentium 4 CPUs at 1.5 to 3.2 GHz. We use the Fairplay run-time system, written in Java and running over Sun JRE 1.5, to execute Yao-protocol episodes. The Yao episode in phase one of the Lindell-Pinkas logarithm protocol completely dominates the running time of the entire logarithm computation, making the peformance of Fairplay itself critical.

We cannot address the performance of multiparty computations without giving special attention to the cost of communication. This element is a wildcard, dependent on link quality and sheer propagation delay across the network distance between the parties. We have done most of our experimentation with the communication component trivialized by running both parties on the same machine or on two machines on the same LAN. For a reality check, we did some experimenting with one party at Yale University in New Haven, CT and the other party at Stevens Institute of Technology in Hoboken, NJ, with a 15 ms round-trip messaging time between them. There was no significant difference in performance in Yao computations. Admittedly, this is at a relatively small network distance. But there is another way to look at this. If network distance were really making the communication cost prohibitive, the two parties anxious to accomplish the joint data-mining computation securely could arrange to run the protocol from outposts of theirs housing prepositioned copies of their respective private data, the outposts securely segregated from each other but at a small network distance. From this perspective, and recognizing that the protocols we are considering involve CPU-intensive cryptographic operations, it is meaningful to assess their performance with the communication component minimized.

With the parties running on 3.2 GHz CPUs, and working with a 60-bit modulus, it takes around 5 seconds to run the complete Lindell-Pinkas logarithm computation. In more detail, to accommodate input x of up to 17 bits (≤ 131071), with $k = 3$ terms of the series to be computed in phase 2 (for an absolute error within 0.0112), we generate a circuit of 1497 gates and the computation runs in around 5.0 seconds. With

the same modulus, to accommodate input x of only up to 13 bits ($\leq$ 8191), allowing $k = 4$ terms of the series to be computed in phase 2 (for an absolute error within 0.0044), we generate a circuit of 1386 gates and the computation runs in around 4.9 seconds. Accommodating inputs of only up to 10 bits (≤ 1023), allowing as many $k = 5$ series terms (for an absolute error within 0.0018), the gate count comes down to 1314 and the running time comes down to around 4.8 seconds.

Clearly, a 5-second wait for a single result of a Lindell-Pinkas secure-logarithm computation seems quite tolerable, but it serves little purpose in itself, of course. This is a shares-to-shares protocol intended for incorporation in a larger data-mining protocol that will ultimately leave the parties with meaningful results. It is reasonable to ask, in such a larger hybrid-SMC protocol, how badly would a 5-second delay for each logarithm computation—and, presumably, comparable delays for other needed SMC building blocks—bog down the entire data-mining algorithm?

We can give a rough idea, based on experiment, of the performance that appears to be possible now in an entire privacy-preserving data-mining computation based on a hybrid-SMC approach. Without fully qualifying the tasks, software versions, and hardware involved, our secure Bayes-net structure-discovery implementation has run against an arbitrarily privately partitioned database of 100,000 records of six fields in about 2.5 hours. This involved almost 500 invocations of the secure logarithm protocol, each involving a Yao-protocol episode run using the Fairplay system, as well as other component protocols. The overall time, computing against this many records, was dominated not by the Yao protocol episodes of the logarithm and minindex components but rather by the scalar-product computations needed to determine securely the numbers of records matching patterns across the private portions of the logical database. The scalar-product computations require a number of homomorphic-encryption operations linear in the number of records in the database.

In developing and using these tools over some time, we note that the room for improvement in performance as implementations are optimized is large. Improvements that do not affect complexity classes, hence of lesser interest to theoreticians, are very significant to practitioners. Improvements in complexity class are there as well; we gained a log factor in our gate counts in the logarithm circuits over our initial naive implementation. Meanwhile, it is clear that significant hybrid-SMC computations are already implementable in a maintainable, modular manner with a development effort that is not exorbitant. Performance of such computations is becoming quite reasonable for realistic application in privacy-preserving data-mining contexts.

Acknowledgments

We thank Benny Pinkas for helpful discussion of the design of the original Lindell-Pinkas logarithm protocol.

References

[KRWF05] Onur Kardes, Raphael S. Ryger, Rebecca N. Wright, and Joan Feigenbaum. Implementing privacy-preserving Bayesian-net discovery for vertically partitioned data. In *Proceedings of the ICDM Workshop on Privacy and Security Aspects of Data Mining*, pages 26–34, 2005.

[LP00] Yehuda Lindell and Benny Pinkas. Privacy preserving data mining. In *Advances in Cryptology – CRYPTO '00*, volume 1880 of *Lecture Notes in Computer Science*, pages 36–54. Springer-Verlag, 2000.

[LP02] Yehuda Lindell and Benny Pinkas. Privacy preserving data mining. *Journal of Cryptology*, 15(3):177–206, 2002.

[MNPS04] Dahlia Malkhi, Noam Nissan, Benny Pinkas, and Yaron Sella. Fairplay – a secure two-party computation system. In *Proc. of the 13th Symposium on Security*, pages 287–302. Usenix, 2004.

[YW06] Zhiqiang Yang and Rebecca N. Wright. Privacy-preserving computation of Bayesian networks on vertically partitioned data. *IEEE Transactions on Data Knowledge Engineering*, 18(9), 2006. An earlier version appeared in KDD 2004.

Constrained *k*-Anonymity: Privacy with Generalization Boundaries

John Miller[*] Alina Campan[*, $] Traian Marius Truta[*]

Abstract: In the last few years, due to new privacy regulations, research in data privacy has flourished. A large number of privacy models were developed most of which are based on the *k*-anonymity property. Because of several shortcomings of the *k*-anonymity model, other privacy models were introduced (*l*-diversity, *p*-sensitive *k*-anonymity, (α, *k*) – anonymity, *t*-closeness, etc.). While differing in their methods and quality of their results, they all focus first on masking the data, and then protecting the quality of the data as a whole. We consider a new approach, where requirements on the amount of distortion allowed to the initial data are imposed in order to preserve its usefulness. Our approach consists of specifying quasi-identifiers generalization boundaries, and achieving *k*-anonymity within the imposed boundaries. We think that limiting the amount of generalization when masking microdata is indispensable for real life datasets and applications. In this paper, the *constrained k-anonymity* model and its properties are introduced and an algorithm for generating constrained *k*-anonymous microdata is presented. Our experiments have shown that the proposed algorithm is comparable with existing algorithms used for generating *k*-anonymity with respect to results quality, and that by using existing unconstrained *k*-anonymization algorithms the generalization boundaries are violated. We also discuss how the constrained *k*-anonymity model can be easily extended to other privacy models.

1 Introduction

A huge interest in data privacy has been generated recently within the public and media [14], as well as in the legislative body [6] and research community.

Many research efforts have been directed towards finding methods to anonymize datasets to satisfy the *k*-anonymity property [16, 17]. These methods also consider minimizing one or more cost metrics between the initial and released microdata (a dataset where each tuple corresponds to one individual entity). Of particular interest are the cost metrics that quantify the *information loss* [2, 5, 19, 27]. Although producing the optimal solution for the *k*-anonymity problem w.r.t. various proposed cost measures has been proved to be NP-hard [9], there are several polynomial algorithms that produce good solutions for the *k*-anonymity problem for real life datasets [1, 2, 8, 9, 21].

Recent results have shown that *k*-anonymity fails to protect the privacy of individuals in all situations [12, 20, 26]. Several privacy models that extend the *k*-anonymity model have been proposed in the literature to avoid *k*-anonymity short-comings: *p*-sensitive *k*-anonymity [20] with its extension called extended *p*-sensitive *k*-anonymity [3], *l*-diversity [12], (α, *k*)-anonymity [24], *t*-closeness [10], (*k*, *e*)-anonymity [28], (*c*, *k*)-safety [13], *m*-confidentiality [25], personalized privacy [26], etc.

In general, the existing anonymization algorithms use different quasi-identifiers generalization strategies in order to obtain a masked microdata that is *k*-anonymous (or satisfies an extension of *k*-anonymity) and conserves as much information intrinsic to the initial microdata as possible. To our knowledge, a privacy model that considers the specification of the maximum allowed generalization level for quasi-identifier attributes in the masked microdata does not exist, nor does a corresponding anonymization algorithm capable of controlling the generalization amount. The ability to limit the amount of allowed generalization could be valuable, and, in fact, indispensable for real life datasets. For example, for some specific data analysis tasks, available masked microdata with the address information generalized beyond the US state level could be useless. In this case the only solution would be to ask the owner of the initial microdata to have the anonymization algorithm applied repeatedly on that data, perhaps with a decreased level of anonymity (a smaller *k*) until the masked microdata satisfies the maximum generalization level requirement (i.e. no address is generalized further than the US state).

In this paper, we first introduce a new anonymity model, called ***constrained k-anonymity***, which preserves the *k*-anonymity requirement while specifying quasi-identifiers generalization boundaries

P3DM'08, April 26, 2008, Atlanta, Georgia, USA.

[*] Department of Computer Science, Northern Kentucky University, U.S.A., {millerj10, campana1, trutat1}@nku.edu

[$] Visiting from Department of Computer Science, Babes-Bolyai University, Romania

(or limits). Second, we describe an algorithm to transform a microdata set such that its corresponding masked microdata will comply with the constrained *k*-anonymity. This algorithm relies on several properties stated and proved for the proposed privacy model.

The paper is organized as follows. Section 2 introduces basic data privacy concepts; and generalization and tuple suppression techniques as a mean to achieve data privacy. Section 3 presents the new constrained *k*-anonymity model. An anonymization algorithm to transform microdata to comply with constrained *k*-anonymity is described in Section 4. Section 5 contains comparative quality results, in terms of information loss, processing time, for our algorithm and one of the existing *k*-anonymization algorithms. The paper ends with future work directions and conclusions.

2 *K*-Anonymity, Generalization and Suppression

Let IM be the initial microdata and MM be the released (a.k.a. masked) microdata. The attributes characterizing IM are classified into the following three categories:

- *identifier* attributes such as *Name* and *SSN* that can be used to identify a record.
- *key* or *quasi-identifier* attributes such as *ZipCode* and *Age* that may be known by an intruder.
- *sensitive* or *confidential* attributes such as *PrincipalDiagnosis* and *Income* that are assumed to be unknown to an intruder.

While the identifier attributes are removed from the published microdata, the quasi-identifier and confidential attributes are usually released to the researchers / analysts. A general assumption is that the values for the confidential attributes are not available from any external source. This assumption guarantees that an intruder cannot use the confidential attributes' values to increase his/her chances of disclosure, and, therefore, modifying this type of attributes values is unnecessary. Unfortunately, an intruder may use record linkage techniques [23] between quasi-identifier attributes and external available information to glean the identity of individuals from the masked microdata. To avoid this possibility of disclosure, one frequently used solution is to modify the initial microdata, more specifically the quasi-identifier attributes values, in order to enforce the *k*-anonymity property.

To rigorously and succinctly express the *k*-anonymity property, we use the following concept:

Definition 1 (*QI-Cluster*): Given a microdata M, a ***QI-cluster*** consists of all the tuples with identical combination of quasi-identifier attribute values in M.

There is no consensus in the literature over the term used to denote a *QI*-cluster. This term was not defined when *k*-anonymity was introduced [17, 18]. More recent papers use different terminologies such as *equivalence class* [24] and *QI-group* [26].

We define *k*-anonymity based on the minimum size of all *QI*-clusters.

Definition 2 (***K-Anonymity Property***): The ***k-anonymity property*** for an MM is satisfied if every *QI*-cluster from MM contains *k* or more tuples.

A general method widely used for masking initial microdata to conform to the *k*-anonymity model is the generalization of the quasi-identifier attributes. Generalization consists in replacing the actual value of the attribute with a less specific, more general value that is faithful to the original [18].

Initially, this technique was used for *categorical* attributes and employed predefined domain and value generalization hierarchies [18]. Generalization was extended for *numerical* attributes either by using *pre-defined hierarchies* [7] or a *hierarchy-free model* [9].

To each categorical attribute a *domain generalization hierarchy* is associated. The values from different domains of this hierarchy are represented in a tree called *value generalization hierarchy*. We illustrate domain and value generalization hierarchy in Figure 1 for attributes *ZipCode* and *Sex*.

There are several ways to perform generalization. Generalization that maps all values of a quasi-identifier categorical attribute from IM to a more general domain in its domain generalization hierarchy is called *full-domain generalization* [9, 16]. Generalization can also map an attribute's values to different domains in its domain generalization hierarchy, each value being replaced by the same generalized value in the entire dataset [7]. The least restrictive generalization, called *cell level generalization* [11], extends Iyengar model [7] by allowing the same value to be mapped to different generalized values, in distinct tuples.

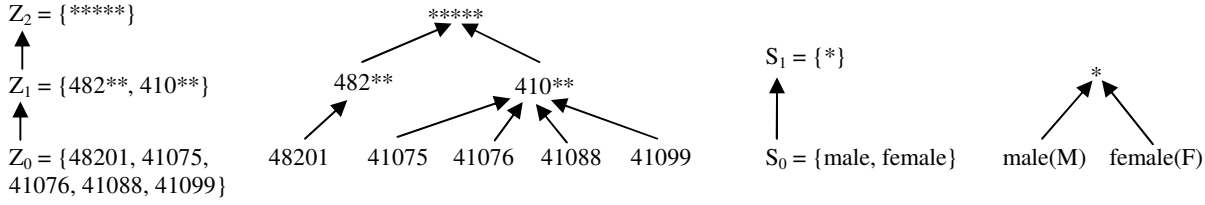


Figure 1: Examples of domain and value generalization hierarchies.

Tuple suppression [16, 18] is the only other method used in this paper for masking the initial microdata. By eliminating entire tuples we are able to reduce the amount of generalization required for achieving the *k*-anonymity property in the remaining tuples. Since the constrained *k*-anonymity model uses generalization boundaries, for many initial microdata sets suppression has to be used in order to generate constrained *k*-anonymous masked microdata.

3 Constrained *K*-Anonymity

In order to specify a generalization boundary, we introduce the concept of a maximum allowed generalization value that is associated with each possible quasi-identifier attribute value from IM. This concept is used to express how far the owner of the data thinks that the quasi-identifier's values could be generalized, such that the resulted masked microdata would still be useful. Limiting the amount of generalization for quasi-identifier attribute values is a necessity for various uses of the data. The data owner is often aware of the way various researchers are using the data and, as a consequence, he/she is able to identify maximum allowed generalization values. For instance, when the released microdata is used to compute various statistical measures related to the US states, the data owner will select the states as maximal allowed generalization values. The desired protection level should be achieved with minimal changes to the initial microdata IM. However, minimal changes may cause generalization that surpasses the maximal allowed generalization values and the masked microdata MM would become unusable. More changes are preferred in this situation if they do not contradict the generalization boundaries.

At this stage, for simplicity, we use predefined hierarchies for both categorical and numerical quasi-identifier attributes, when defining maximal allowed generalization values. Techniques to dynamically build hierarchies for numerical attributes exist in the literature [4] and we intend to use them in our future research.

Definition 3. (***Maximum Allowed Generalization Value***): Let Q be a quasi-identifier attribute (categorical or numerical), and $\mathcal{H}_Q$ its predefined value generalization hierarchy. For every leaf value $v \in \mathcal{H}_Q$, the ***maximum allowed generalization value*** of v, denoted by $MAGVal(v)$, is the value (leaf or not-leaf) in $\mathcal{H}_Q$ situated on the path from v to the root, such that:

- for any released microdata, the value v is permitted to be generalized only up to $MAGVal(v)$ and
- when several *MAGVal*s exist on the path between v and the hierarchy root, then the $MAGVal(v)$ is the first *MAGVal* that is reached when following the path from v to the root node.

Figure 2 contains an example of defining maximal allowed generalization values for a subset of values for the *Location* attribute. The *MAGVal*s for the leaf values "San Diego" and "Lincoln" are "California", and, respectively, "Midwest" (the *MAGVal*s are marked by * characters that delimit them). This means that the quasi-identifier *Location*'s value "San Diego" may be generalized to itself or "California", but not to "West Coast" or "United States". Also, "Lincoln" may be generalized to itself, "Nebraska", or "Midwest", but not to "United States".

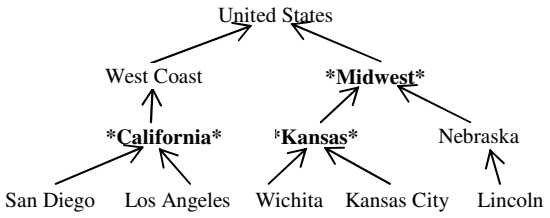


Figure 2: Examples of *MAGVal*s.

The second requirement in the *MAGVal*'s definition specifies that the hierarchy path between a leaf value v and $MAGVal(v)$ can contain no node other than $MAGVal(v)$ that is a maximum allowed generalization value. This restriction is imposed in order to avoid any ambiguity about the *MAGVal*s of the leaf values in a sensitive attribute hierarchy. Note that several *MAGVal*s may exist on a path between a leaf and the root as a result of defining *MAGVal*s for other leaves within that hierarchy.

Definition 4. (***Maximum Allowed Generalization Set***): Let Q be a quasi-identifier attribute and $\mathcal{H}_Q$ its predefined value generalization hierarchy. The set of all *MAGVal*s for attribute Q is called Q's ***maximum allowed generalization set***, and it is denoted by $MAGSet(Q) = \{ MAGVal(v) \mid \forall v \in leaves(\mathcal{H}_Q) \}$ (The notation $leaves(\mathcal{H}_Q)$ represents all the leaves from the $\mathcal{H}_Q$ value generalization hierarchy).

Given the hierarchy for the attribute *Location* presented in Figure 2, $MAGSet(Location) = \{$California, Kansas, Midwest$\}$.

Usually, the data owner/user only has generalization restrictions for some of the quasi-identifiers in a microdata that is to be masked. If for a particular quasi-identifier attribute Q there are not any restrictions in respect to its generalization, then no maximal allowed generalization values are specified for Q's value hierarchy; in this case, each leaf value in $\mathcal{H}_Q$ is considered to have the $\mathcal{H}_Q$'s root value as its maximal allowed generalization value.

Record	Name	SSN	Age	Location	Sex	Race	Diagnosis	Income
r_1	Alice	123456789	32	San Diego	M	W	AIDS	17,000
r_2	Bob	323232323	30	Los Angeles	M	W	Asthma	68,000
r_3	Charley	232345656	42	Wichita	M	W	Asthma	80,000
r_4	Dave	333333333	30	Kansas City	M	W	Asthma	55,000
r_5	Eva	666666666	35	Lincoln	F	W	Diabetes	23,000
r_6	John	214365879	20	Lincoln	M	B	Asthma	55,000
r_7	Casey	909090909	25	Wichita	F	B	Diabetes	23,000

Figure 3: An initial microdata set $\mathcal{IM}$

Record	Age	Location	Sex	Race
r_1	30-32	California	M	W
r_2	30-32	California	M	W
r_3	30-42	MidWest	*	W
r_4	30-42	MidWest	*	W
r_5	30-42	MidWest	*	W
r_6	20-25	MidWest	*	B
r_7	20-25	MidWest	*	B

a)

Record	Age	Location	Sex	Race
r_1	30-32	California	M	W
r_2	30-32	California	M	W
r_3	25-42	Kansas	*	*
r_4	25-42	Kansas	*	*
r_7	25-42	Kansas	*	*
r_5	20-35	Lincoln	*	*
r_6	20-35	Lincoln	*	*

b)

Figure 4: Two masked microdata sets $\mathcal{MM}_1$ and $\mathcal{MM}_2$ for the initial microdata $\mathcal{IM}$. (Only the quasi-identifier attribute values are shown in the masked microdata sets)

Definition 5. (***Constraint Violation***): We say that the masked microdata $\mathcal{MM}$ has a ***constraint violation*** if one quasi-identifier value, v, in $\mathcal{IM}$, is generalized in one tuple in $\mathcal{MM}$ beyond its specific maximal generalization value, $MAGVal(v)$.

Definition 6. (***Constrained K-Anonymity***): The masked microdata $\mathcal{MM}$ satisfies the ***constrained k-anonymity property*** if it satisfies k-anonymity and it does not have any constraint violation.

We note that a k-anonymous masked microdata may have multiple constraint violations, but any masked microdata that satisfies constrained k-anonymity property will not have any constraint violations; or in other words, any quasi-identifier value, v, from the initial microdata will never be generalized beyond its $MAGVal(v)$ in any constrained k-anonymous masked microdata.

Consider the following example. The initial microdata set $\mathcal{IM}$ in Figure 3 is characterized by the following attributes: *Name* and *SSN* are identifier attributes (to be removed from the masked microdata), *Age*, *Location*, *Sex*, and *Race* are the quasi-identifier attributes, and *Diagnosis* and *Income* are the sensitive attributes. The attribute *Location* values and their *MAGVal*s are described by Figure 2. The remaining quasi-identifier attributes do not have any generalization boundary requirements.

Figure 4 illustrates two possible masked microdata $\mathcal{MM}_1$ and $\mathcal{MM}_2$ for the initial microdata $\mathcal{IM}$. In this figure, only quasi-identifier values are shown, the confidential attribute values will be kept unchanged from the initial microdata $\mathcal{IM}$ (*Diagnosis* and *Income* attributes from Figure 3). The first masked microdata, $\mathcal{MM}_1$, satisfies 2-anonymity, but contradicts constrained 2-anonymity w.r.t. *Location* attribute's maximal allowed generalization. On the other hand, the second microdata set, $\mathcal{MM}_2$, satisfies constrained 2-anonymity: every *QI*-cluster consists of at least 2 tuples, and none of the *Location* initial attribute's values are generalized beyond its *MAGVal*.

4 *GreedyCKA* - An Algorithm for Constrained *K*-Anonymization

In this section we assume that the initial microdata set $\mathcal{IM}$, the generalization boundaries for its quasi-identifier attributes, expressed as *MAGVal*s in their corresponding hierarchies, and the k value (as in k-anonymity) are given. First, we will describe a method to decide if $\mathcal{IM}$ can be masked to comply with constrained k-anonymity using generalization only, and second, we will introduce an algorithm for achieving constrained k-anonymity.

Our approach to constrained k-anonymization partially follows an idea found in [1] and [2], which consists in modeling and solving k-anonymization as a clustering problem. Basically, the algorithm takes an initial microdata set $\mathcal{IM}$ and establishes a "good" partitioning of it into clusters. The released microdata set $\mathcal{MM}$ is afterwards formed by generalizing the quasi-identifier attributes' values of all tuples inside each cluster to the same values (called generalization information for a cluster). However, it is not always possible to mask an initial microdata to satisfy constrained k-anonymity only by generalization. Sometimes a solution to constrained k-anonymization has to combine generalization with suppression. In this case, our algorithm suppresses the *minimal* set of tuples

from IM such that is possible to build a constrained k-anonymous masked microdata for the remaining tuples.

The constrained k-anonymization by clustering problem can be formally stated as follows.

Definition 7. (***Constrained K-Anonymization by Clustering Problem***): Given a microdata IM, the ***constrained k-anonymization by clustering problem*** for IM is to find a partition $\mathcal{S} = \{cl_1, cl_2, \ldots, cl_v, cl_{v+1}\}$ of IM, where $cl_j \subseteq IM$, j=1..v+1, are called clusters and: $\bigcup_{j=1}^{v} cl_j = IM - cl_{v+1}$; $cl_i \bigcap cl_j = \emptyset$, $i, j = 1..v+1$, $i \neq j$; $|cl_j| \geq k$, j=1..v ; and a cost measure is optimized. The cluster cl_{v+1} is formed of all the tuples in IM that have to be suppressed in MM, and the tuples within every cluster cl_j, j=1..v will be generalized (their quasi-identifier attributes) in MM to common values.

The generalization information of a cluster, which is introduced next, represents the minimal covering "tuple" for that cluster. Since in this paper we use predefined value generalization hierarchies for both categorical and numerical attributes, we do not have to consider a definition that distinguishes between these two types of attributes [21].

Definition 8. (***Generalization Information***): Let $cl = \{r_1, r_2, \ldots, r_u\}$ be a cluster of tuples selected from IM, $QI = \{Q_1, Q_2, \ldots, Q_s\}$ be the set of quasi-identifier attributes. The ***generalization information of cl*** w.r.t. quasi-identifier attribute set QI is the "tuple" $gen(cl)$, having the scheme QI, where for each attribute $Q_j \in QI$, $gen(cl)[Q_j]$ = the lowest common ancestor in $\mathcal{H}_{Q_j}$ of $\{r_1[Q_j], \ldots, r_u[Q_j]\}$.

For the cluster cl, its generalization information $gen(cl)$ is the tuple having as value for each quasi-identifier attribute the most specific common generalized value for all that attribute values from *cl's* tuples. In the corresponding MM, each tuple from the cluster cl will have its quasi-identifier attributes values replaced by $gen(cl)$.

To decide whether an initial microdata can be masked to satisfy constrained k-anonymity property using generalization only, we introduce several properties. These properties will also allow us, in case that constrained k-anonymity cannot be achieved using generalization only, to select the tuples that must be suppressed.

Property 1. Let IM be a microdata set and cl a cluster of tuples from IM. If cl contains two tuples r_i and r_j such that $MAGVal(r_i[Q]) \neq MAGVal(r_j[Q])$, where Q is a quasi-identifier attribute, then the generalization of the tuples from cl to $gen(cl)$ will create at least one constraint violation.

Proof. Assume that there are two tuples r_i and r_j within cl such that $MAGVal(v_i) \neq MAGVal(v_j)$, where $v_i = r_i[Q]$ and $v_j = r_j[Q]$, $v_i, v_j \in leaves(\mathcal{H}_Q)$. Let a be a value within H_Q that is the first common ancestor for $MAGVal(v_i)$ and $MAGVal(v_j)$. Depending on how $MAGVal(v_i)$ and $MAGVal(v_j)$ are located relatively to one another in the Q's value generalization hierarchy, a can be one of them, or a value on a superior tree level. In any case, a will be different from, and an ancestor for at least one of $MAGVal(v_i)$ or $MAGVal(v_j)$. This is a consequence of the fact that $MAGVal(v_i) \neq MAGVal(v_j)$: a common ancestor of two different nodes x and y in a tree is a node which is different from at least one of the nodes x and y. Because of this fact, when cl will be generalized to $gen(cl)$, $gen(cl)[Q]$ will be a (or depending on the other tuples in cl, even an ancestor of a) – therefore at least one of the values v_i and v_j will be generalized further than its maximal allowed generalization value, leading to a constraint violation. // q.e.d.

Property 1 restricts the possible solutions of the constrained anonymization by clustering problem to those partitions $\mathcal{S}$ of IM for which every cluster to be generalized doesn't show any constraint violation w.r.t. each of the quasi-identifier attributes. The following definition introduces a masked microdata that will help us to express when the IM can be transformed to satisfy constrained k-anonymity using generalization only.

Definition 9. (***Maximum Allowed Microdata***): The ***maximum allowed microdata*** for a microdata IM, MAM, is the masked microdata where every quasi-identifier value, v, in IM is generalized to $MAGVal(v)$.

Property 2. For a given IM, if its maximum allowed microdata MAM is not k-anonymous, then any masked microdata obtained from IM by applying generalization only will not satisfy constrained k-anonymity.

Proof. Assume that MAM is not k-anonymous, and there is a masked microdata MM that satisfies constrained k-anonymity. This means that every QI-cluster from MM has at least k elements and it does not have any constraint violation. Let cl_i be a cluster of elements from IM that is generalized to a QI-cluster in MM (i = 1, .., v). Because MM satisfies constrained k-anonymity, the generalization of cl_i to $gen(cl_i)$ does not create any constraint violation. Based on Property 1, for each quasi-identifier attribute, all entities from cl_i share the same $MAGVal$s. As a consequence, by generalizing all quasi-identifier attributes values to their corresponding $MAGVal$s (this is the procedure to create the MAM microdata) all entities from the cluster cl_i (for all i = 1, .., v) will be contained within the same QI-cluster. This

means that each *QI*-cluster in $\mathcal{MAM}$ contains one or more *QI*-clusters from $\mathcal{MM}$ and its size will, then, be at least *k*. In conclusion, $\mathcal{MAM}$ is *k*-anonymous, which is a contradiction with our initial assumption. // q.e.d.

Property 3. If $\mathcal{MAM}$ satisfies *k*-anonymity then $\mathcal{MAM}$ satisfies the constrained *k*-anonymity property.

Proof. This follows from the definition of $\mathcal{MAM}$.

Property 4. An initial microdata, $\mathcal{IM}$, can be masked to comply with constrained *k*-anonymity using only generalization if and only if its corresponding $\mathcal{MAM}$ satisfies *k*-anonymity.

Proof. "If": If $\mathcal{MAM}$ satisfies *k*-anonymity, then based on Property 3, $\mathcal{MAM}$ is also constrained *k*-anonymous, and $\mathcal{IM}$ can be masked to $\mathcal{MAM}$ (in the worst case – or even to a less generalized masked microdata) to comply with constrained *k*-anonymity.

"Only If": If $\mathcal{MAM}$ does not satisfy *k*-anonymity, then based on Property 2, any masked microdata obtained by applying generalization only to $\mathcal{IM}$ will not satisfy constrained *k*-anonymity. // q.e.d.

Now we have all the tools required to check whether an initial microdata $\mathcal{IM}$ can be masked to satisfy the constrained *k*-anonymity property using generalization only. We follow the next two steps:

- Compute $\mathcal{MAM}$ for $\mathcal{IM}$. This is done by replacing each quasi-identifier attribute value with its corresponding *MAGVal*.
- If all *QI*-clusters from $\mathcal{MAM}$ have at least *k* entities than the $\mathcal{IM}$ can be masked to satisfy constrained *k*-anonymity.

It is very likely that there are some *QI*-clusters in $\mathcal{MAM}$ with size less than *k*. We use the notation $\mathcal{OUT}$ to represent all entities from these *QI*-clusters (for simplicity we use the same notation to refer to entities from both $\mathcal{IM}$ and $\mathcal{MAM}$). Unfortunately, the entities from $\mathcal{OUT}$ cannot be *k*-anonymized while preserving the constraint condition, as shown by the Property 6. For a given $\mathcal{IM}$ with its corresponding $\mathcal{MAM}$ and $\mathcal{OUT}$ sets the following two properties hold:

Property 5. $\mathcal{IM} \setminus \mathcal{OUT}$ can be masked using generalization only to comply with constrained *k*-anonymity.

Proof. By definition of the $\mathcal{OUT}$ set, all *QI*-clusters from $\mathcal{MAM} \setminus \mathcal{OUT}$ have size *k* or more, which means that $\mathcal{MAM} \setminus \mathcal{OUT}$ satisfies the *k*-anonymity property. Based on Property 4 ($\mathcal{MAM} \setminus \mathcal{OUT}$ is the maximum allowed microdata for $\mathcal{IM} \setminus \mathcal{OUT}$), $\mathcal{IM} \setminus \mathcal{OUT}$ can be masked using generalization only to comply with constrained *k*-anonymity. // q.e.d.

Property 6. Any subset of $\mathcal{IM}$ that contains one or more entities from $\mathcal{OUT}$ cannot be masked using generalization only to achieve constrained *k*-anonymity.

Proof. We assume that there is an initial microdata $\mathcal{IM}'$, a subset of $\mathcal{IM}$, that contains one or more entities from $\mathcal{OUT}$, and $\mathcal{IM}'$ can be masked using generalization only to comply with constrained *k*-anonymity. Let $x \in \mathcal{OUT} \cap \mathcal{IM}'$. Let $\mathcal{MAM}'$ be the maximum allowed microdata for $\mathcal{IM}'$. Based on Property 4, if $\mathcal{IM}'$ can be masked to be constrained *k*-anonymous, then $\mathcal{MAM}'$ is *k*-anonymous, therefore *x* will belong to a *QI*-cluster with size at least *k*. By construction $\mathcal{MAM}'$ is a subset of $\mathcal{MAM}$, and therefore, the size of each *QI*-cluster from $\mathcal{MAM}$ is equal to or greater than the size of the corresponding *QI*-cluster from $\mathcal{MAM}'$. This means that *x* will belong to a *QI*-cluster with size at least *k* in the $\mathcal{MAM}$. This is a contradiction with $x \in \mathcal{OUT}$. // q.e.d.

The Properties 5 and 6 show that $\mathcal{OUT}$ is the minimal tuple set that must be suppressed from $\mathcal{IM}$ such that the remaining set could be constrained *k*-anonymized. To compute a constrained *k*-anonymous masked microdata using minimum suppression and generalization only we follow an idea found in [1] and [2], which consists in modeling and solving *k*-anonymization as a clustering problem. First, we suppress all tuples from the $\mathcal{OUT}$ set. Next, we create all *QI*-clusters in the maximum allowed microdata for $\mathcal{IM} \setminus \mathcal{OUT}$. Last, each such cluster will be divided further, if possible, using the clustering approach from [1, 2], into several clusters, all with size greater than or equal to *k*. This approach uses a greedy technique that tries to optimize an information loss (*IL*) measure. The information loss measure we use in our algorithm implementation was introduced in [2]. We present it in Definitions 10 and 11. Note that this *IL* definition assumes that value generalization hierarchies are predefined for all quasi-identifier attributes.

Definition 10. (***Cluster Information Loss***): Let $cl \in \mathcal{S}$ be a cluster, $gen(cl)$ its generalization information and $QI = \{Q_1, Q_2, .., Q_t\}$ the set of quasi-identifier attributes. The ***cluster information loss*** caused by generalizing *cl* tuples to $gen(cl)$ is:

$$IL(cl) = |cl| \cdot \sum_{j=1}^{t} \frac{height(\Lambda(gen(cl)[Q_j]))}{height(H_{Q_j})}$$

where:

- $|cl|$ denotes the cluster *cl* cardinality;
- $\Lambda(w)$, $w \in H_{Qj}$ is the subhierarchy of H_{Qj} rooted in *w*;
- $height(H_{Qj})$ is the height of the tree hierarchy H_{Qj}.

Definition 11. (***Total Information Loss***)**:** ***Total information loss*** for a partition $\mathcal{S}$ of the initial microdata set is the sum of the information loss measure for all clusters in $\mathcal{S}$.

In is worth noting that, for the constrained k-anonymization by clustering problem, the cluster of tuples to be suppressed, cl_{v+1}, will have the maximum possible *IL* value for a cluster of the same size as cl_{v+1}. The information loss for this cluster will be: $IL(cl_{v+1}) = |cl_{v+1}| \cdot n$, where n is the number of quasi-identifier attributes. When performing experiments to compare the quality of constrained k-anonymous microdata and k-anonymous microdata, produced for the same IM, the information loss of the constrained k-anonymous solution includes the information loss caused by the suppressed cluster as well, and not only for the generalized clusters. More than that, for every suppressed tuple we consider the maximum information loss that it can cause when it is masked. This way, the quality of the constrained k-anonymous solutions will not be biased because of a favored way of computing information loss for the suppressed tuples.

The two-stage constrained k-anonymization algorithm called *GreedyCKA* is depicted in Figure 5.

We present below the pseudocode of the *GreedyCKA* Algorithm:

```
Algorithm GreedyCKA is
Input   IM – initial microdata;
        k – as in k-anonymity;
Output  S={cl1,cl2,… clv,clv+1} – a solution for
        the constrained k-anonymization by
        clustering problem for IM;
 Compute MAM and OUT;
 S = ∅;
 For each QI-cluster from MAM \ OUT, cl,
 {
   // By cl we refer to the entities from IM
   // that are clustered together in MAM.
   S' = Greedy_k-member_Clustering(cl, k); //[2]
   S = S ∪ S';
 }
 v = |S|;
 clv+1 = OUT;
End GreedyCKA;
```

This idea of dividing IM into clusters based on common *MAGVal*s of the quasi-identifiers can be employed for other privacy models as well, not only for k-anonymity. For instance, if we use an algorithm that creates a p-sensitive k-anonymous masked microdata [20], such as *EnhancedPKClustering* [22], we just need to execute that algorithm instead of *Greedy_k-member_Clustering*, for each *QI*-cluster from $MAM \setminus OUT$. The obtained masked microdata will be p-sensitive k-anonymous and will satisfy the generalization boundaries. We can define this new privacy model as ***constrained p-sensitive k-anonymity***. Using similar modifications in the *GreedyCKA* algorithm, we can introduce constrained versions of other privacy models such as: ***constrained l-diversity*** [12], ***constrained t-closeness*** [10], etc. and generate their corresponding masked microdata sets.

5 Experimental Results

In this section we compare the *GreedyCKA* and *Greedy_k-member_Clustering* [2] algorithms with respect to: the quality of the results they produce measured against the information loss measure; the algorithms' efficiency as expressed by their running time; the number of constraint violation that k-anonymous masked microdata produced by *Greedy_k-member_Clustering* have; and the suppression amount performed by *GreedyCKA* in order to produce constrained k-anonymous masked microdata in presence of different constraint sets.

The two algorithms were implemented in Java; tests were executed on a dual CPU machine with 3.00 GHz and 1 GB of RAM, running Windows 2003 Server.

A set of experiments were performed for an IM consisting of 10,000 tuples randomly selected from the *Adult* dataset from the UC Irvine Machine Learning Repository [15]. In all the experiments, we considered a set of eight quasi-identifier attributes: *education-num*, *workclass*, *marital-status*, *occupation*, *race*, *sex*, *age*, and *native-country*.

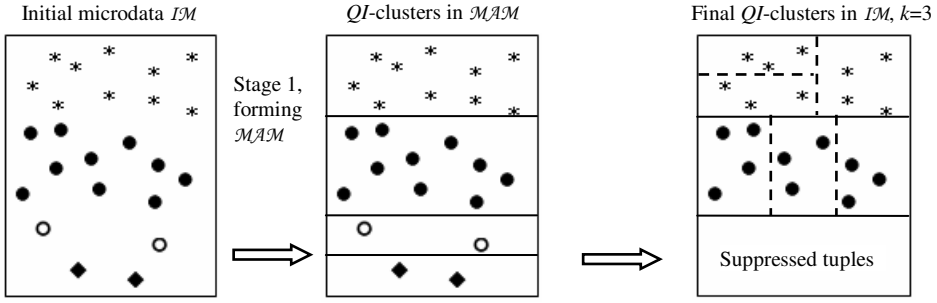


Figure 5: The two-stage process in creating constrained k-anonymous masked microdata.

The *GreedyCKA* and *Greedy_k-member_Clustering* algorithms were applied to this microdata set, for different k values, from k=2 to k=10. Two different generalization constraint sets were successively considered for every k value. First, only the *native-country* attribute's values were subject to generalization constraints, as depicted in Figure 6. Second, both *native-country* and *age* had generalization boundaries; the value generalization hierarchy and the maximum allowed generalization values for the *age* attribute are illustrated in Figure 7. In Figures 6 and 7, the *MAGVal*s are depicted as bold and delimited by * characters. Of course, *Greedy_k-member_Clustering* proceeded without taking into consideration the generalization boundaries, as it is a "simple", unconstrained k-anonymization algorithm. This is why the masked microdata it produces will generally contain numerous constraint violations. On the other side, the k-anonymization process of *GreedyCKA* is conducted in respect to the specified generalization boundaries; this is why the masked microdata produced by *GreedyCKA* is free of constraint violations.

The quasi-identifier attributes without generalization boundaries have the following heights for their corresponding value generalization hierarchies: *education-num* – 4, *workclass* – 1, *marital-status* – 2, *occupation* – 1, *race* – 1, and *sex* – 1.

However, masking microdata to comply with the more restrictive constrained k-anonymity model sometimes comes with a price. As the experiments show, it is possible to lose more of the intrinsic microdata information when masking it to satisfy constrained k-anonymity than when masking it to satisfy k-anonymity only. Figure 8 presents comparatively the information loss measure for the masked microdata created by *GreedyCKA* and *Greedy_k-member_Clustering*, with the two different constraint sets and for k values in the range 2-10.

As expected, the information loss value is generally greater when constraints are considered in the k-anonymization process. Exceptions may however occur. For example, *GreedyCKA* obtained better results then *Greedy_k-member_Clustering* for k = 8, 9 and 10, when only *native_country* was constrained. The information lost is influenced, of course, by the constraint requirements and by the microdata distribution w.r.t. the constrained attributes. When more quasi-identifiers have generalization boundaries or more restrictive generalization boundaries, the information lost in the constrained k-anonymization process will generally increase.

Regarding the running time, we can state that *GreedyCKA* will always be more efficient than *Greedy_k-member_Clustering*. The explanation for this fact is that, when generalization boundaries are imposed, they will cause the initial microdata to be divided in several subsets (the *QI*-clusters of $\mathcal{MAM}$), on which *Greedy_k-member_Clustering* will be afterwards applied. *Greedy_k-member_Clustering* has an $O(n^2)$ complexity, and applying it on smaller microdata subsets will reduce the processing time. More constraints and *QI*-clusters exist in $\mathcal{MAM}$, more significant is the reduction of the processing time for microdata masking (see Figure 9).

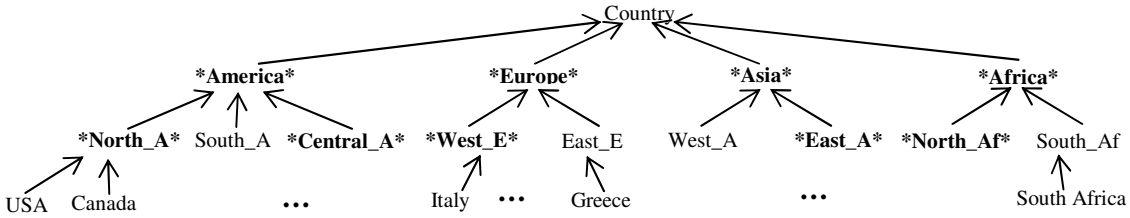


Figure 6: *MAGVals* for the quasi-identifier attribute *Country*.

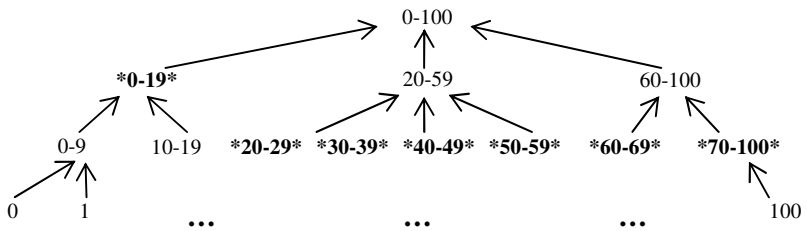


Figure 7: *MAGVals* for the quasi-identifier attribute *Age*.

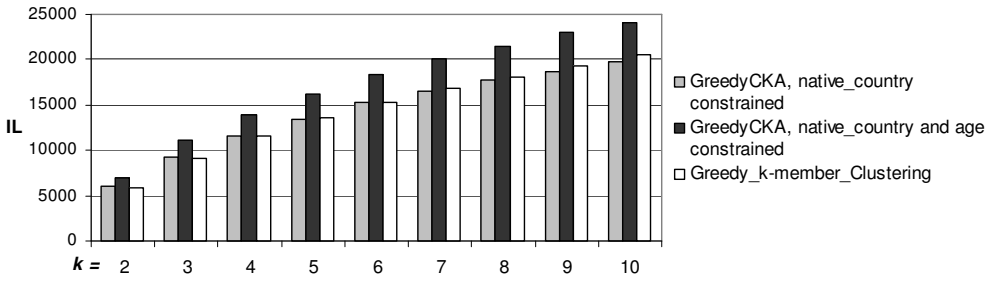


Figure 8: Information Loss (*IL*) for *GreedyCKA* and *Greedy_k-member_Clustering*.

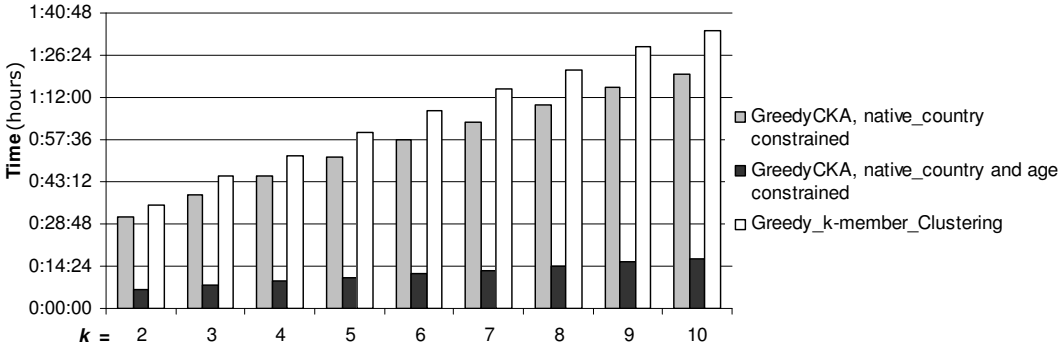


Figure 9: Running Time for *GreedyCKA* and *Greedy_k-member_Clustering*.

As pointed out, when *Greedy_k-member_Clustering* is applied to *k*-anonymize *IM*, the resulting masked microdata usually contains numerous constraint violations. Table 1 reports the number of constraint violations in the outcome of the *Greedy_k-member_Clustering* unconstrained *k*-anonymization algorithm, for two maximal generalization requirement sets.

k	No of constraint violations for 1 constrained attribute – **native_country**	No of constraint violations for 2 constrained attributes – **native_country, age**
2	605	2209
3	991	3824
4	1377	5297
5	1657	6163
6	1906	6964
7	2198	7743
8	2354	8417
9	2550	8931
10	2728	9549

Table 1: Constraint violations in *Greedy_k-member_Clustering*

k	2	3	4	5	6	7	8	9	10
No of suppressed tuples for 1 constrained attribute – **native_country**	0	0	0	0	0	0	0	0	0
No of suppressed tuples for 2 constrained attributes– **native_country, age**	5	15	24	28	48	60	81	97	106

Table 2: Number of tuples suppressed by *GreedyCKA*

Table 2 shows the number of tuples suppressed by *GreedyCKA*, while masking the initial microdata.

All in all, our experiments proved that constrained *k*-anonymous masked microdata can be achieved without sacrificing the data quality to a significant extent, when compared to a corresponding *k*-anonymous unconstrained masked microdata.

While the constrained *k*-anonymity model responds to a necessity in real-life applications, the existing *k*-anonymization algorithms are not able to build masked microdata that comply with it. In this context, *GreedyCKA* takes optimal suppression decisions, based on the proved properties of the new model (Properties 5 and 6), and builds high-quality constrained *k*-anonymous masked microdata.

6 Conclusions and Future Work

In this paper we defined a new privacy model, called constrained *k*-anonymity, which takes into consideration generalization boundaries imposed by the data owner for quasi-identifier attributes. Based on the model properties, an efficient algorithm to generate a masked microdata to comply with constrained *k*-anonymity property was introduced. Our experiments showed that the proposed algorithm obtains comparable information loss values with *Greedy_k-member_Clustering* algorithm, while the masked microdata sets obtained by the latter have many constraint violations.

In this paper we used predefined hierarchies for all quasi-identifier attributes. As future work we plan to extend this concept further for numerical attributes. We plan to provide a technique to dynamically determine for each numerical quasi-identifier value, its maximal allowed generalization, based on that attribute's values in the analyzed microdata and a minimal user input.

We also pointed out that the constraint *k*-anonymity property and even our proposed algorithm, *GreedyCKA*, can be extended to other privacy models (models such as constrained *l*-diversity, constrained (α, *k*)-anonymity, constrained *p*-sensitive *k*-anonymity, etc. can be easily defined). Finding specific properties for these enhanced privacy models, and developing improved algorithms to generate masked microdata to comply with such models are subject of future work.

Acknowledgments

This work was partially supported by the CNCSIS (Romanian National University Research Council) grant PNCDI-PN II, IDEI 550/2007.

References

[1] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu, *Achieving Anonymity via Clustering*, in Proc. of the ACM PODS (2006), pp. 153–162.

[2] J.W. Byun, A. Kamra, E. Bertino, and N. Li, *Efficient k-Anonymization using Clustering Techniques*, in Proc. Of DASFAA (2006), pp. 188–200.

[3] A. Campan, T. M. Truta, *Extended P-Sensitive K-Anonymity*, Studia Universitatis Babes-Bolyai, Informatica, Vol. 51, No. 2 (2006), pp. 19–30.

[4] B. C. M. Fung, K. Wang, and P. S. Yu, *Anonymizing classification data for privacy preservation*, IEEE Transactions on Knowledge and Data Engineering, Vol. 19, No. 5 (2007), pp. 711–725.

[5] G. Ghinita, K. Karras, P. Kalinis, and N. Mamoulis, *Fast Data Anonymization with Low Information Loss*, in Proc. of VLDB (2007), pp. 758–769.

[6] HIPAA, *Health Insurance Portability and Accountability Act*, www.hhs.gov/ocr/hipaa, 2002.

[7] V. Iyengar, *Transforming Data to Satisfy Privacy Constraints*, in Proc. of the ACM SIGKDD (2002), pp. 279–288.

[8] K. LeFevre, D. DeWitt, and R. Ramakrishnan, *Incognito: Efficient Full-Domain K-Anonymity*, in Proc. of the ACM SIGMOD (2005), pp. 49–60.

[9] K. LeFevre, D. DeWitt, and R. Ramakrishnan, *Mondrian Multidimensional K-Anonymity*, in Proc. of the IEEE ICDE (2006), pp. 25.

[10] N. Li, T. Li, and S. Venkatasubramanian, *T-Closeness: Privacy Beyond k-Anonymity and l-Diversity*, in Proc. of the IEEE ICDE (2007), pp. 106–115.

[11] M. Lunacek, D. Whitley, and I. Ray, *A Crossover Operator for the k-Anonymity Problem*, in Proc. of the GECCO (2006) pp. 1713–1720.

[12] A. Machanavajjhala, J. Gehrke, and D. Kifer, *L-Diversity: Privacy beyond K-Anonymity*, in Proc. of the IEEE ICDE (2006), pp. 24.

[13] D. J. Martin, D. Kifer, A. Machanavajjhala, and J. Gehrke, *Worst-Case Background Knowledge for Privacy-Preserving Data Publishing*, Proc. of the IEEE ICDE (2007), pp. 126–135.

[14] MSNBC, *Privacy Lost*, www.msnbc.msn.com/id/ 15157222 , 2006.

[15] D. J. Newman, S. Hettich, C. L. Blake, and C. J. Merz, *UCI Repository of Machine Learning Databases*, www. ics.uci.edu/~mlearn/MLRepository.html, 1998.

[16] P. Samarati, *Protecting Respondents Identities in Microdata Release*, IEEE Transactions on Knowledge and Data Engineering, Vol. 13, No. 6 (2001), pp. 1010–1027.

[17] L. Sweeney, *k-Anonymity: A Model for Protecting Privacy*, International Journal on Uncertainty, Fuzziness, and Knowledge-based Systems, Vol. 10, No. 5 (2002), pp. 557–570.

[18] L. Sweeney, *Achieving k-Anonymity Privacy Protection Using Generalization and Suppression*, International Journal on Uncertainty, Fuzziness, and Knowledge-based Systems, Vol. 10, No. 5 (2002), pp. 571–588.

[19] T. M. Truta, F. Fotouhi, and D. Barth-Jones, *Privacy and Confidentiality Management for the Microaggregation Disclosure Control Method*, in Proc. of the PES Workshop, with ACM CCS (2003), pp. 21–30.

[20] T. M. Truta, V. Bindu, *Privacy Protection: P-Sensitive K-Anonymity Property*, in Proc. of the PDM Workshop, with IEEE ICDE (2006), pp. 94.

[21] T. M. Truta, A. Campan, *K-Anonymization Incremental Maintenance and Optimization Techniques*, in Proc. of the ACM SAC (2007), pp. 380–387.

[22] T. M. Truta, A. Campan, P. Meyer, *Generating Microdata with P-Sensitive K-Anonymity Property*, in Proc. of the SDM Workshop, with VLDB (2007), pp. 124–141.

[23] W. Winkler, *Matching and Record Linkage*, Business Survey Methods, Wiley (1995), pp. 374–403.

[24] R. C. W. Wong, J. Li, A. W. C. Fu, and K. Wang, *(α, k)-Anonymity: An Enhanced k-Anonymity Model for Privacy-Preserving Data Publishing*, in Proc. of the ACM SIGKDD (2006), pp. 754–759.

[25] R. C. W. Wong, J. Li, A. W. C. Fu, and J. Pei, *Minimality Attack in Privacy-Preserving Data Publishing*, in Proc. of the VLDB (2007), pp. 543–554.

[26] X. Xiao, Y. Tao, *Personalized Privacy Preservation*, in Proc. of the ACM SIGMOD (2006), pp. 229–240.

[27] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. Fu, *Utility-Based Anonymization Using Local Recoding*, in Proc. of ACM SIGKDD (2006), pp. 785–790.

[28] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu, *Aggregate Query Answering on Anonymized Tables*, in Proc. Of the IEEE ICDE (2007), pp. 116–125.

Privacy-Preserving Predictive Models for Lung Cancer Survival Analysis

Glenn Fung[1], Shipeng Yu[1], Cary Dehing-Oberije[2], Dirk De Ruysscher[2], Philippe Lambin[2], Sriram Krishnan[1], R. Rao Bharat[1]
[1] CAD and Knowledge Solutionis, Siemens Medical Solutions USA, Inc., Malvern, PA USA.
[2] MAASTRO clinic, the Netherlands.

Abstract

Privacy-preserving data mining (PPDM) is a recent emergent research area that deals with the incorporation of privacy preserving concerns to data mining techniques. We consider a real clinical setting where the data is horizontally distributed among different institutions. Each one of the medical institutions involved in this work provides a database containing a subset of patients. There is recent work that shows the potential of the PPDM approach in medical applications. However, there is few work in developing/implementing PPDM for predictive personalized medicine. In this paper we use real data from several institutions across Europe to build models for survival prediction for non-small-cell lung cancer patients while addressing the potential privacy preserving issues that may arise when sharing data across institutions located in different countries. Our experiments in a real clinical setting show that the privacy preserving approach may result in improved models while avoiding the burdens of traditional data sharing (legal and/or anonymization expenses).

1 Introduction

Privacy-preserving data mining (PPDM) is a recent emergent research area that deals with the incorporation of privacy preserving concerns to data mining techniques. We are particularly interested in a scenario when the data is horizontally distributed among different institutions. In the medical domain this means that each medical institution (hospitals, clinics, etc.) provides a database containing a complete (or almost complete) subset of item sets (patients). An efficient PPDM algorithm should be able to process the data from all the sources and learn data mining/machine learning models that take into account all the information available without sharing explicitly private information among the sources. The ultimate goal of a PPDM model is to perform similarly or identically to a model learned by having access to all the data at the same time.

P3DM'08, April 26, 2008, Atlanta, Georgia, USA.

There are have been a push for the incorporation of electronic health records (EHR) in medical institutions worldwide. There seems to be a consensus that the availability of EHR will have several significant benefits for health systems across the world, including: improvement of quality of care by tracking performance on clinical measures, better and more accurate insurance reimbursement, computer assisted diagnosis (CAD) tools, etc. Therefore, there is a constant increase on the number of hospitals saving huge amounts of data that can be used to build predictive models to assist doctors in the medical decision process for treatment, diagnosis, and prognosis among others. However, sharing the data across institutions becomes a difficult and tedious process that also involves considerable legal and economic burden on the institutions sharing the medical data.

In this paper we explore two privacy preserving techniques applied to learn survival predictive models for non-small-cell lung cancer patients treated with (chemo) radiotherapy. We use real data collected from patients treated on three European institutions in two different countries (the Netherlands and Belgium) to build our models. The framework we are describing in this paper allows to design/learn improved predictive models that perform better than the individual models obtained by using local data from only one institution, without addressing the local and international privacy preserving concerns that arise when sharing patient-related data. As far as we know, there is none previous work related to learning survival models for lung cancer radiation therapy addressing PP concerns.

The rest of the paper is organized as follows: in the next section, we introduced the notation used in the paper. In section 3 we present an overview of the related work. In sections 4.1 and 4.3 we present the overview of the two methods used for our predictive models: Newton-Lagrangian Support Vector Machines [5] and Cox Regression [3]. Later in sections 4.2 and 4.4, we present the technical details of the corresponding privacy preserving (PP) algorithms used. We conclude

the paper describing our application with experimental results performed in a real clinical setting and the conclusions.

2 Notation

We describe our notations now. All vectors will be column vectors unless transposed to a row vector by a prime $'$. For a vector $x \in R^n$ the notation x_j will signify either the j-th component or j-th block of components. The scalar (inner) product of two vectors x and y in the n-dimensional real space R^n will be denoted by $x'y$. The notation $A \in R^{m\times n}$ will signify a real $m \times n$ matrix. For such a matrix, A' will denote the transpose of A, A_i will denote the i-th row or i-th block of rows of A. A vector of ones in a real space of arbitrary dimension will be denoted by e. Thus for $e \in R^m$ and $y \in R^m$ the notation $e'y$ will denote the sum of the components of y. A vector of zeros in a real space of arbitrary dimension will be denoted by 0. For $A \in R^{m\times n}$ and $B \in R^{k\times n}$, a *kernel* $K(A, B')$ maps $R^{m\times n} \times R^{n\times k}$ into $R^{m\times k}$. In particular, if x and y are column vectors in R^n then, $K(x', y)$ is a real number, $K(x', B')$ is a row vector in R^k and $K(A, B')$ is an $m \times k$ matrix. The abbreviation "s.t." stands for "subject to".

3 Related Work

As a consequence of the recent advances of network computing, there has been recently great interest in privacy-preserving data mining techniques. An extensive review of PPDM techniques can be found in [14]. Most of the available data mining techniques require and assume that there is complete access to all data at all times. This may not be true for example, in an uncentralized distributed medical setting where for each data source or institution, there are local procedures in place to enforce privacy and security of the data. If this is the case, there is a need to use efficient data mining and machine learning techniques that can use data across institutions while complying with the non-disclosure nature of the available data. There are two main kinds of data partitioning when dealing with distributed setting where PPDM is needed: a) the data is partitioned vertically, this means that all institutions have some subset of features (predictors, variables) for all the available patients. When this is the case, several techniques have been proposed to address the issue including: adding random perturbations to the data [2, 4]. The other popular PPDM setting occurs when the data is partitioned horizontally among institutions, that means that different entities hold the same input features for different groups of individuals. This case have been addressed in [16, 15] by privacy-preserving SVMs and induction tree classifiers. There are several other recently proposed privacy preserving classifying techniques including cryptographically private SVMs [7], wavelet-based distortion [10]. There is recent work that shows the potential of the approach [6, 12] in medical settings. However , there is few work in developing/implementing PPDM for predictive personalized medicine.

4 Privacy-Preserving Predictive Models (PPPM)

In this section we introduce two PP predictive models, namely PP Support Vector Machines and PP Cox Regression. We first give an overview of the two techniques in sections 4.1 and 4.3, and then present the PP versions in sections 4.2 and 4.4.

4.1 Overview of Support Vector Machines. We describe in this section the fundamental classification problems that lead to the standard quadratic Support vector machine (SVM) formulation that minimizes a quadratic convex function. We consider the problem of classifying m points in the n-dimensional real space R^n, represented by the $m \times n$ matrix A, according to membership of each point A_i in the classes +1 or -1 as specified by a given $m \times m$ diagonal matrix D with ones or minus ones along its diagonal. For this problem, the standard support vector machine with a linear kernel AA' [13] is given by the following quadratic program for some $\nu > 0$:

$$\begin{array}{rl} \min\limits_{(w,\gamma,y)\in R^{n+1+m}} & \nu e'y + \frac{1}{2}w'w \\ \text{s.t.} & D(Aw - e\gamma) + y \geq e \\ & y \geq 0. \end{array} \tag{4.1}$$

As depicted in Figure 1, w is the normal to the bounding planes:

$$\begin{array}{rcrcl} x'w & - & \gamma & = & +1 \\ x'w & - & \gamma & = & -1, \end{array} \tag{4.2}$$

and γ determines their location relative to the origin. The first plane above bounds the class +1 points and the second plane bounds the class -1 points when the two classes are strictly linearly separable, that is when the slack variable $y = 0$. The linear separating surface is the plane

$$x'w = \gamma, \tag{4.3}$$

midway between the bounding planes (4.2). If the classes are linearly inseparable then the two planes bound the two classes with a "soft margin" determined by a nonnegative slack variable y, that is:

$$\begin{array}{rcrcll} x'w - \gamma & + & y_i & \geq & +1, & \text{for } x' = A_i \text{ and } D_{ii} = +1, \\ x'w - \gamma & - & y_i & \leq & -1, & \text{for } x' = A_i \text{ and } D_{ii} = -1. \end{array} \tag{4.4}$$

The 1-norm of the slack variable y is minimized with weight ν in (4.1). The quadratic term in (4.1), which is twice the reciprocal of the square of the 2-norm distance $\frac{2}{\|w\|}$ between the two bounding planes of (4.2) in the n-dimensional space of $w \in R^n$ for a *fixed* γ, maximizes that distance, often called the "margin". Figure 1 depicts the points represented by A, the bounding planes (4.2) with margin $\frac{2}{\|w\|}$, and the separating plane (4.3) which separates $A+$, the points represented by rows of A with $D_{ii} = +1$, from $A-$, the points represented by rows of A with $D_{ii} = -1$. For this paper we used Newton-

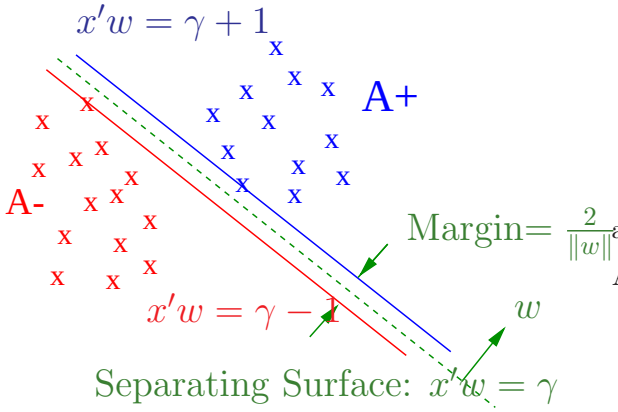


Figure 1: **The bounding planes (4.2) with margin $\frac{2}{\|w\|}$, and the plane (4.3) separating $A+$, the points represented by rows of A with $D_{ii} = +1$, from $A-$, the points represented by rows of A with $D_{ii} = -1$.**

Lagrangian SVM (NSVM), an algorithm based on an essentially equivalent formulations of this classification problem [5]. In this formulation, the square of 2-norm of the slack variable y is minimized with weight $\frac{\nu}{2}$ instead of the 1-norm of y as in (4.1). In addition the distance between the planes (4.2) is measured in the $(n+1)$-dimensional space of $(w, \gamma) \in R^{n+1}$, that is $\frac{2}{\|(w,\gamma)\|}$. Measuring the margin in this $(n+1)$-dimensional space instead of R^n induces strong convexity and has little or no effect in general on the problem.

4.2 Privacy Preserving SVMs. For our privacy preserving application we chose to use a technique on random kernel mappings recently proposed by Mangasarian and Wild on [11]. The algorithm is based on two simple basic ideas:

1. **The use of reduced kernel mappings** [9, 8], where the kernel centers are randomly chosen. Instead of using the complete kernel function $K(A, A') : R^{m\times n} \rightarrow R^{m\times m}$ as it is usually done in kernel methods they propose the use of a reduced kernel $K(A, B') : R^{m\times n} \rightarrow R^{m\times \tilde{m}}$, where $B \in R^{\tilde{m}\times n}$ is a completely random matrix with fewer rows than the number of available features, $(\tilde{m} < n)$.
2. **Each entity makes public only a common randomly generated linear transformation of the data** given by the matrix product of its privately held matrix of data rows multiplied by the transpose of a common random matrix B for linear kernels, and a similar kernel function for nonlinear kernels. In our experimental setting, we assumed that all the available patient data is normalized between 0 and 1 and therefore the elements of B were generated according to a normal distribution with mean zero, variance one and standard deviation one.

Next, we formally introduce the PPSVM algorithm as presented in [11]

ALGORITHM 4.1. **Nonlinear PPSVM Algorithm**

(I) All q entities agree on the same random matrix $B \in R^{\bar{m}\times n}$ with $\bar{m} < n$ for security reasons as justified in the explanation immediately following this algorithm. All entities make public the class matrix D (labels) where $D_{ll} = \pm, l = 1, \ldots, m$ for the each of the data matrices $A_i, i = 1, \ldots, q$ that they all hold.

(II) Each entity generates its own privately held random matrix $B_{.j} \in R^{\bar{m}\times n_j}$, $j = 1, \ldots\ldots, p$, where n_j is the number of input features held by entity j.

(III) Each entity j makes public its nonlinear kernel $K(A_j, B')$. This does not reveal A_j but allows the public computation of the full nonlinear kernel:
(4.5)

$$K(A, B') = K\left(\begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_q \end{bmatrix}, B'\right) = \begin{bmatrix} K(A_1, B') \\ K(A_2, B') \\ \vdots \\ K(A_q, B') \end{bmatrix}$$

(IV) A publicly calculated linear classifier $K(x', B')u - \gamma = 0$ is computed by any linear hyperplane based classification or regression method method such as the ones presented in sections 4.1 and 4.3.

(V) For each new *$x \in R^n$, obtained by an entity, that entity privately computes $K(x', B')$ and classifies the given x according to the sign of $K(x', B')u - \gamma$.*

Note that algorithm 4.1 works for any kernel with the following associative property:

$$K\left(\begin{bmatrix} C \\ D \end{bmatrix}, F\right) = \begin{bmatrix} K(C, F) \\ K(D, F) \end{bmatrix}$$

Which is, in particular, the case of the linear kernel $K(A, B') = AB'$ and that we will use for the rest of the paper.

As stated in [11], it is important to note than in the the above algorithm no entity j reveals its data nor its components of a new testing data point. When $\bar{m} < n$, there is an infinite number of matrices $A_i \in R^{mi \times n}$ in the solution set of the equation $A_i B' = P_i$, when B and P_i are given. This claim can be justified by the well-known properties of under-determined systems of linear equations. Furthermore, the following proposition which is originally stated and proved in [11] is aimed to formally support the claim presented above:

PROPOSITION 4.2. **(infinite solutions of $A_i B' = P_i$ if $\bar{m} < n$)** *Given the matrix product $P_i' = AiB' \in R^{m_i \times \bar{m}}$, where $A_i \in R^{m_i \times n}$ is unknown and B is a known matrix in $R^{\bar{m} \times n}$ with $\bar{m} < n$, there are an infinite number of solutions, including:*

$$\begin{pmatrix} n \\ \bar{m} \end{pmatrix}^{m_i} = \left(\frac{n!}{(n-\bar{m})!\bar{m}!} \right)^{m_i}$$

possible solutions $A_i \in R^{m_i \times n}$ to the equation $A_i B' = P_i$. Furthermore, the infinite number of matrices in the affine hull of these $\begin{pmatrix} n \\ \bar{m} \end{pmatrix}^{m_i}$ matrices also satisfy $A_i B' = P_i$.

4.3 Overview of Cox Regression. Cox regression, or the Cox propositional-hazards model, is one of the most popular algorithms for survival analysis [3]. Apart from baing a classification algorithm which directly deal with binary or multi-class outcomes, Cox regression defines a semi-parametric model to directly relate the predictive variables with the real outcome which is in general the survival time (e.g., in years).

Let T represent survival time. The so-called *hazard function* is a representation of the distribution of survival times, which assesses the instantaneous risk of demise at time t, conditional on survival to that time:

$$h(t) = \lim_{\Delta t \to 0} \frac{\Pr[(t \leq T < t + \Delta t) | T \geq t]}{\Delta t}.$$

The Cox regression model assumes a linear model for the log-hazard, or as a multiplicative model for the hazard:

$$\log h(t) = \alpha(t) + w'x, \tag{4.6}$$

where x denote the covariates for each observation, and the *baseline hazard* $\alpha(t)$ is unspecified. This model is semi-parametric because while the baseline hazard can take any form, the covariates enter the model linearly. Now given any two observations x_i and x_j, from the definition of hazard function we can get

$$\frac{h(t_i)}{h(t_j)} = \exp[w'(x_i - x_j)],$$

which is independent of time t. The baseline hazard $\alpha(t)$ also does not affect the hazard ratio. This is why the Cox model is a *proportional-hazards model*.

And Cox has showed in [3] that even though the baseline hazard is unspecified, the Cox model can still be estimated by the method of *partial likelihood*. It is also possible to extract an estimate of the baseline hazard after having fit the model.

4.4 Privacy Preserving Cox Regression. The main idea of the privacy preserving SVM is to perform a random mapping of the original predictive variables into a new space, and then perform standard SVM on the new space. Since in the Cox regression the interaction between the parameter of the models and the data is linear, we can also apply the same idea presented in section 4.2 for the *privacy preserving Cox regression*. Given the random matrix B and assuming that we are using a linear kernel, equation 4.6 is slightly changed to:

$$\log h(t) = \alpha(t) + w'xB', \tag{4.7}$$

Again it is important to note, that to our knowledge, this is the first time that privacy preserving techniques are applied for survival analysis methods.

5 Application: 2-Year Survival Prediction for Non-Small Cell Lung Cancer Patients

Radiotherapy, combined with chemotherapy, is treatment of choice for a large group of non-small cell lung cancer (NSCLC) patients. The treatment is not restricted to patients with mediastinal lymph node metastasis, but is also indicated for patients who are inoperable because of their physical condition. In addition, the marginal role of radiotherapy and chemotherapy for the survival of NSCLC patients has been changed into one of significant importance. Improved radiotherapy treatment techniques allow an increase of the radiation dose, while at the same time more effective chemoradiation schemes are being applied. These developments have lead to an improved outcome in terms of survival. Although the introduction of FDG-PET scans has enabled more accurate detection of positive lymph nodes and distant metastases, leading to stage migration, the TNM staging system is still highly inaccurate for the prediction of survival outcome for this group of patients [1]. In summary, an increasing number of patients is being treated successfully with (chemo) radiation, but

an accurate estimation of the survival probability for an individual patient, taking into account patient, tumor as well as treatment characteristics and offering the possibility for treatment decision-making, is currently not available.

At present, generally accepted prognostic factors for inoperable patients are performance status, weight loss, presence of comorbidity, use of chemotherapy in addition to radiotherapy, radiation dose and tumor size. For other factors such as gender and age the literature shows inconsistent results, making it impossible to draw definitive conclusions. In these studies CT-scans were used as the major staging tool. However, the increasing use of FDG-PET scans offers the possibility to identify and use new prognostic factors. In a recent study it was shown that number of involved nodal areas quantified by PET-CT was an important prognostic factor [1]. We performed this retrospective study to develop and validate several prediction models for 2-year survival of NSCLC patients, treated with (chemo) radiotherapy, taking into account all known prognostic factors. To the best of our knowledge, this is the first study of prediction models for NSCLC patients treated with (chemo)radiotherapy

5.1 Patient Population. Between May 2002 and January 2007, a total number of 455 inoperable NSCLC patients, stage I-IIIB, were referred to MAASTRO clinic to be treated with curative intent. Clinical data of all these patients were collected retrospectively by reviewing the clinical charts. If PET was not used as a staging tool, patients were excluded from the study. This resulted in the inclusion of 399 patients. The primary gross tumor volume ($\mathrm{GTV_{primary}}$) and nodal gross tumor volume ($\mathrm{GTV_{nodal}}$) were calculated, as delineated by the treating radiation oncologist, using a commercial radiotherapy treatment planning system (Computerized Medical Systems, Inc, CMS). The sum of $\mathrm{GTV_{primary}}$ and $\mathrm{GTV_{nodal}}$ resulted in the GTV. For patients treated with sequential chemotherapy these volumes were calculated using the post-chemotherapy imaging information. The creation of the volumes was based on PET and CT information only; bronchoscopic findings were not taken into account. The number of positive lymph node stations was assessed by the nuclear medicine specialist using either an integrated FDG-PET-CT scan or a CT-scan combined with FDG-PET-scan. T-stage and N-stage were assessed using pre-treatment CT, PET and mediastinoscopy when applicable. For patients treated with sequential chemotherapy stage as well as number of positive lymph node stations was assessed using pre-chemotherapy imaging information.

Additionally, a smaller number of patients treated at the other two centers, the Gent hospital and the Leuven hospital, were also collected for this study. There are respectively 112 and 40 patients from the Gent and Leuven hospitals, and the same set of clinical variables as the MAASTRO patients were measured.

5.2 Radiotherapy Treatment Variables. No elective nodal irradiation was performed and irradiation was delivered 5 days per week. Radiotherapy planning was performed with a Focus (CMS) system, taking into account lung density and according to ICRU 50 guidelines. There were four different radiotherapy treatment regimes applied for these patients in this retrospective study, therefore to account for the different treatment time and number of fractions per day, the equivalent dose in 2 Gy fractions, corrected for overall treatment time (EQD2,T), was used as a measure for the intensity of chest radiotherapy 5.8. Adjustment for dose per fraction and time factors were made as follows:

$$\mathrm{EQD2,T} = D\left(\frac{d+\beta}{2+\beta}\right) - \gamma \max(0, T - T_k), \tag{5.8}$$

where D is the total radiation dose, d is dose per fraction, $\beta = 10$ Gy, T is overall treatment time, T_k is the accelerated repopulation kick-off time which is 28 days, and γ is the loss in dose per day due to repopulation which is 0.66 Gy/day.

5.3 Experimental Setup. In this paper we focus on 2-year survival prediction for these NSCLC patients, which is the most interesting prediction from clinical perspective. The survival status was evaluated in December 2007. The following 6 clinical predictors are used to build the prediction models: gender (two groups: male/female), WHO performance status (three groups: $0/1/\geq 2$), lung function prior to treatment (forced expiratory volume, in the range of $17 \sim 139$), number of positive lymph node stations (five groups: $0/1/2/3/\geq 4$), natural logarithm of GTV (in the range of $-0.17 \sim 6.94$), and the equivalent dose corrected by time (EQD2,T) from (5.8). The mean values across patients are used to impute the missing entries if some of these predictors are missing for certain patients. To account for the very different number of patients from the three sites, a subset of MAASTRO patients were selected for the following study. In the following we use the names "MAASTRO", "Gent" and "Leuven" to denote the data from the three different centers.

For the SVM methods, since they can only deal with binary outcome, we only use the patients with 2-year follow-up and create an outcome for them with $+1$ meaning they survived 2 years, and -1 meaning they didn't survive 2 years. This setting leads to 70, 37 and

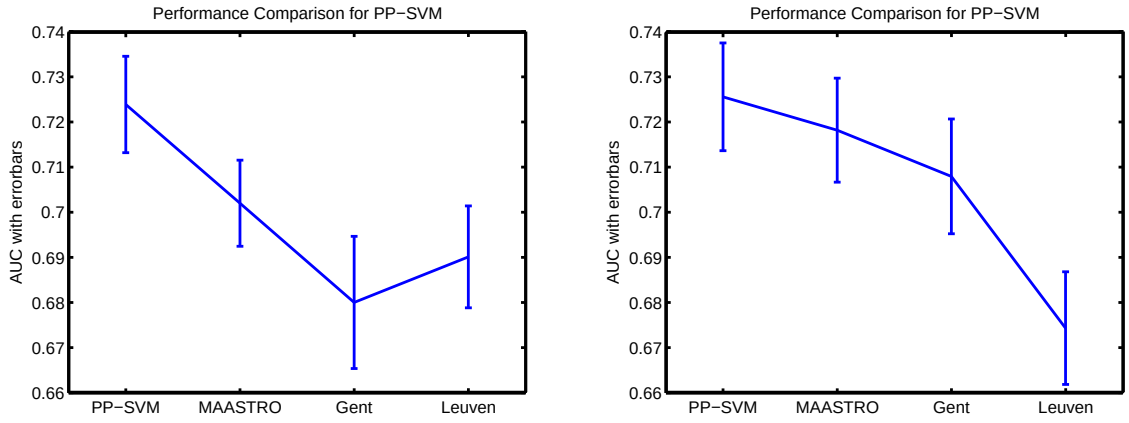


Figure 2: AUC comparison for privacy preserving SVMs with 40% (left) and 60% (right) training patients. The error bars are calculated based on 100 times of random splits of the data.

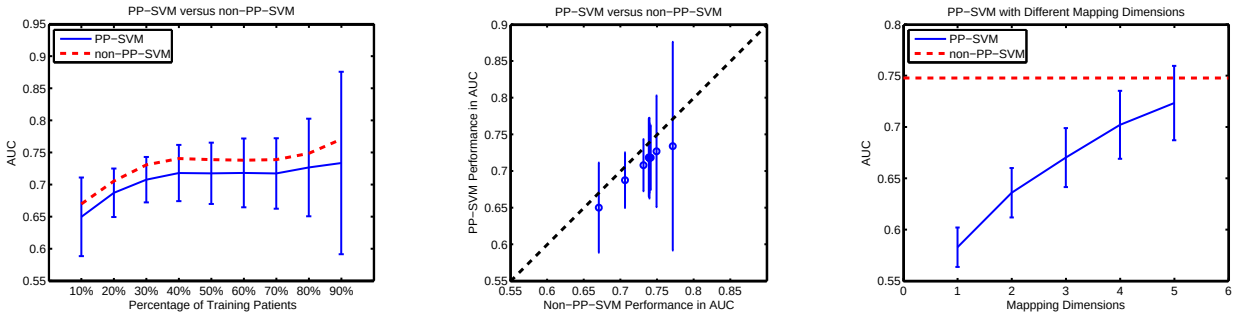


Figure 3: AUC comparison between PP-SVMs and non PP-SVMs (which explicitly use all the training data from different centers, and thus upper-bound the predictive performance of PP-SVMs). We compare the two with different percentages of training patients (left), in a scatter plot (middle), and with different dimensions $\bar{m}$ for PP-SVMs (right) for a 40% split.

23 patients for the MAASTRO, Gent and Leuven sets, respectively. For the Cox regression methods, we can potentially use all the patients with the exact number of survived years, and do right censoring for those patients who are still alive. Under this setting we end up with 80, 85 and 40 patients for MAASTRO, Gent and Leuven, respectively.

Under the privacy preserving setting, we are interested in assessing the predictive performance of a model combining the patient data from the three centers together, compared to the models trained based on each of these centers. The data combination needs to be done in a way that sensitive information is not uncovered. Therefore for our experiments we trained the following 4 models under each configuration:

- **PP model**: Apply the privacy preserving techniques we have introduced and train a model using combined data from the three centers.
- **MAASTRO, Gent and Leuven models**: Train models using only the MAASTRO, Gent and Leuven training patients repectively.

For each of the configurations, we vary the percentage of training patients in each of the centers, and report the Area Under the ROC Curve (AUC) for the test patients. Note that the testing was performed using all the test patients from all centers.

6 Results

In Figure 2 we show the results for privacy preserving SVM models, with 2 example training percentages (40% and 60%). The other percentages yield similar results. The error bars are over 100 runs with random split of training/test patients for each center, and each time a random B matrix of dimensionality 5×6 is used for the PP-SVM models. As can be seen, the PP-SVM models achieve the best performance compared to other single-center based models. This is mainly because PP-SVM models are able to use more data in model training, at

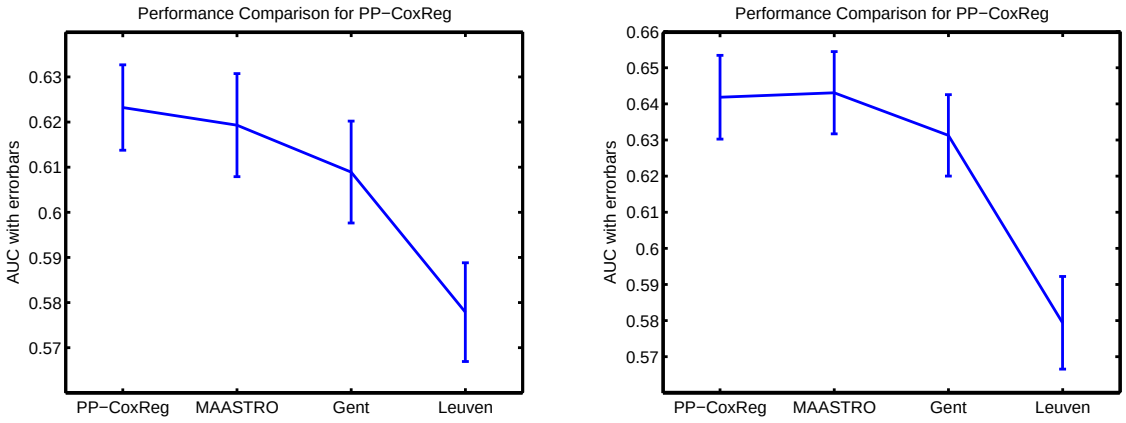


Figure 4: AUC comparison for privacy preserving Cox regression models with 40% (left) and 60% (right) training patients. The error bars are calculated based on 100 times of random splits of the data.

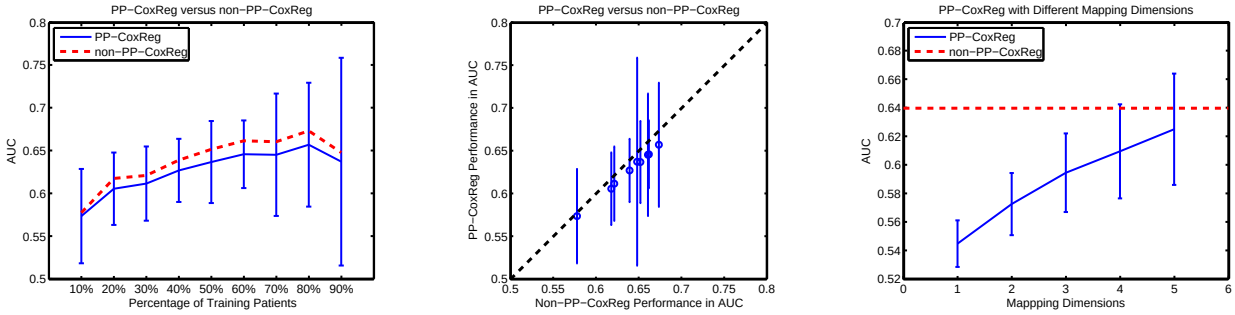


Figure 5: AUC comparison between PP-CoxReg and non PP-CoxReg (which explicitly use all the training data from different centers, and thus upper-bound the predictive performance of PP-CoxReg). We compare the two with different percentages of training patients (left), in a scatter plot (middle), and with different dimensions $\bar{m}$ for PP-CoxReg (right) in a 40% split.

the same time without violating the privacy regulations. When we increase the training percentages, all models will improve (compare Figure 2 right to left), and the single-center based models have a higher improvement. However the PP-SVM models still perform the best.

It is easy to realize that PP-SVM will end up with a performance loss compared to a non PP-SVM model, which explicitly combines all the training patients from different centers and does not preserve privacy. This is because in PP-SVMs a random matrix B projects each patient into a lower dimensional space (for privacy preserving purpose), and thus leads to information loss. To empirically evaluate how much performance loss the PP-SVMs have, we show a more extensive comparison in Figure 3. On the left we show the comparison with different percentages of the training/test splits, and as can be seen the gaps between PP-SVMs and non PP-SVMs are not very big. This indicates PP-SVMs can achieve similar predictive performance while satisfying the privacy preserving requirement. The scatter plot in the middle is another way to visualize these results. On the right we vary the mapping dimensions $\bar{m}$ for the B matrix we used in PP models, and as expected, bigger $\bar{m}$ yield better predictive performance. Therefore, in practice we normally choose $\bar{m} = n - 1$ to maximize the performance of the PP models (which still perfectly satisfies the privacy preserving requirements). From this comparison we see that there is a big error bar for different B matrices, and one interesting future work is to identify the best B matrix for PP models.

In Figure 4 we also empirically evaluate the results for privacy preserving Cox regression models, also with the 2 example training percentages (40% and 60%). They have the same trend as we have seen in Figure 2, but it is interesting that with a higher percentage of training data (e.g., 60% on the right), PP-CoxReg performs the same as the model trained using only MAASTRO training patients. This indicates PP-

CoxReg model is more sensitive to the different characteristics of the data from different centers. In practice, we need to carefully investigate the different data distributions to estimate the benefits of combining them.

We also empirically compare the PP Cox regression models with non PP-CoxReg models in Figure 5. As can be seen, the gaps between PP-CoxReg and non PP-CoxReg models are even smaller than those between PP-SVM and non PP-SVM models, meaning PP-CoxReg models are more accurate toward the non privacy preserving solutions. In practice we still need to choose $\bar{m} = n - 1$ to maximize the PP-CoxReg performance, and to choose the best B matrix if possible.

7 Discussion and Conclusions

We have applied a simple recently proposed PP technique in a real clinical setting where data is shared across three European institutions in order to build more accurate predictive models than the ones obtained using only data from one institute. We have extended the previously proposed PP algorithm (originally suggested for SVM) to cox regression. As far as we know this is the first work that addresses privacy preserving concerns for survival models. The work presented here is based on preliminary results and we are already working on designing improved algorithms to address several concerns that arise when performing our experiments. One of the concerns that arise (as shown in section 6) is how to address the impact of the variability of the matrix B on the performance of the predictive models. For that, we are currently experimenting with formulations in which the B matrix is intended not only to "de-identify" the data but also to optimally improve model performance. Another relevant concern that we are looking into is, how to weight the importance of data from different institutions, assuming that the reliability of the data or the labels varies among institutions.

References

[1] Dehing-Oberije C, De Ruysscher D, van der Weide H, and et al. Tumor volume combined with number of positive lymph node stations is a more important prognostic factor than tnm stage for survival of non-small-cell lung cancer patients treated with (chemo)radiotherapy. *Int J Radiat Oncol Biol Phys*, (in press).

[2] K. Chen and L. Liu. Privacy preserving data classification with rotation perturbation. In *Proceedings of the Fifth International Conference of Data Mining (ICDM'05)*, pages 589–592. IEEE, 2005.

[3] D. R. Cox. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society*, Series B 34:187–220, 1972.

[4] Wenliang Du, Yunghsiang Han, and Shigang Chen. Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, pages 222–233, 2004. http://citeseer.ist.psu.edu/du04privacypreserving.html.

[5] G. Fung and O. L. Mangasarian. Finite Newton method for Lagrangian support vector machine classification. *Special Issue on Support Vector Machines. Neurocomputing*, 55:39–55, 2003.

[6] Gang Kou, Yi Peng, Yong Shi, and Zhengxin Chen. Privacy-preserving data mining of medical data using data separation-based techniques. *Data Science Journal*, 6:429–434, 2007.

[7] S. Laur, H. Lipmaa, and T. Mielikäinen. Cryptographically private support vector machines. In L. Ungar, M. Craven, D. Gunopulos, and T. Eliassi-Rad, editors, *Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 618–624, 2006.

[8] Y.-J. Lee and S.Y. Huang. Reduced support vector machines: A statistical theory. *IEEE Transactions on Neural Networks*, 18:1–13, 2007.

[9] Y.-J. Lee and O. L. Mangasarian. RSVM: Reduced support vector machines. In *Proceedings of the First SIAM International Conference on Data Mining*, 2001.

[10] L. Liu, J. Wang, Z. Lin, and J. Zhang. Wavelet-based data distortion for privacy-preserving collaborative analysis. Technical Report 482-07, Department of Computer Science, University of Kentucky, Lexington, KY 40506, 2007. http://www.cs.uky.edu/ jzhang/pub/MINING/lianliu1.pdf.

[11] O. L. Mangasarian and E. Wild. Privacy-preserving classification of horizontally partitioned data via random kernels. Technical Report 07-03, Computer sciences department, university of Wisconsin - Madison, Madison, WI, 2007.

[12] G. Schadow, S. J. Grannis, and C. J. McDonald. Privacy-preserving distributed queries for a clinical case research network. pages 55–65, 2002.

[13] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, second edition, 2000.

[14] V. Verykios, E. Bertino, I. Fovino, L. Provenza, Y. Saygin, and Y. Theodoridis. State-of-the-art in privacy preserving data mining. *SIGMOD*, 33:50–57, 2004.

[15] Ming-Jun Xiao, Liu-Sheng Huang, Yong-Long Luo, and Hong Shen. Privacy preserving id3 algorithm over horizontally partitioned data. In *PDCAT '05: Proceedings of the Sixth International Conference on Parallel and Distributed Computing Applications and Technologies*, pages 239–243, Washington, DC, USA, 2005. IEEE Computer Society.

[16] Hwanjo Yu, Xiaoqian Jiang, and Jaideep Vaidya. Privacy-preserving svm using nonlinear kernels on horizontally partitioned data. In *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*, pages 603–610, New York, NY, USA, 2006. ACM.