UTILITY INDEPENDENT PRIVACY PRESERVING DATA MINING – HORIZONTALLY PARTITIONED DATA

E Poovammal [1] and M Ponnavaikko [2]*

*[*1]Department of Computer Science and Engineering, SRM University, Chennai – 603203, India*
Email: epsrm@yahoo.com
[2]Bharathidasan University, Trichy- 600024, Tamilnadu, India
Email: ponnav@gmail.com

ABSTRACT

Micro data is a valuable source of information for research. However, publishing data about individuals for research purposes, without revealing sensitive information, is an important problem. The main objective of privacy preserving data mining algorithms is to obtain accurate results/rules by analyzing the maximum possible amount of data without unintended information disclosure. Data sets for analysis may be in a centralized server or in a distributed environment. In a distributed environment, the data may be horizontally or vertically partitioned. We have developed a simple technique by which horizontally partitioned data can be used for any type of mining task without information loss. The partitioned sensitive data at 'm' different sites are transformed using a mapping table or graded grouping technique, depending on the data type. This transformed data set is given to a third party for analysis. This may not be a trusted party, but it is still allowed to perform mining operations on the data set and to release the results to all the 'm' parties. The results are interpreted among the 'm' parties involved in the data sharing. The experiments conducted on real data sets prove that our proposed simple transformation procedure preserves one hundred percent of the performance of any data mining algorithm as compared to the original data set while preserving privacy.

Keywords - Anonymization, Distributed database, Linking attack, Privacy preservation, Quasi identifiers

1 INTRODUCTION

Progress in scientific research depends on the availability and sharing of information and ideas. Micro data (tables with un-aggregated data) are a valuable source of information for research while protecting the privacy of human participants during data sharing is given top priority by researchers. To avoid identification of individuals' information, attributes, such as *name* and *SSN* (Social Security Number), are removed from the table before data release. However, some attributes, such as *age*, *gender*, and *zip code* (called quasi identifiers), can be linked with external data to uniquely identify the individual. For example, a patient data set can be linked with a voter data set by means of the quasi-identifying attributes to discover an individual's diagnosis. This problem is called a linking attack. Therefore, there is a need for privacy preserving data publishing.

Data publishing techniques tend to study different transformation methods associated with privacy. These techniques include methods such as randomization, k-anonymity, and l-diversity. The randomization method is expressed by Adam and Wortmann (1989) and Wenliang and Zhijun, (2003). K-anonymity is introduced by Sweeney (2002), discussed by Aggarwal (2005), and also applied by Aggarwal, Feder, Kenthapadi, Khuller, Motwani, Panigrahy, Thomas, and Zhu (2006). L-diversity is introduced by Machanavajjhala, Gehrke, Kifer, and Venkitasubramaniam (2006). However, randomization methods deal with numerical attributes, whereas k-anonymity and L-diversity focus on categorically sensitive attributes. None of the previous anonymization principles prevents proximity breach, which is a privacy threat specific to numerically sensitive attributes, discussed in Section 2.1. The inadequacy of all these methods in preventing proximity attack is discussed by Li, Tao, and Xiao (2008). However, we have observed a new privacy threat, especially in categorically sensitive attributes, and call it a divergence breach. The proposed transformation technique prevents both proximity and divergence breaches in released data. The data may be stored in a centralized server or in a distributed environment. In a distributed environment, data may be horizontally or vertically partitioned.

1.1 Horizontally Partitioned Data

Consider different hospitals that wish to conduct joint research while preserving the privacy of their patients. In this scenario, it is necessary to protect privileged information, but it is also necessary to enable the use of this information for research or other purposes. Data with the same attributes at different sites are called horizontally partitioned data. A researcher can mine very useful rules and patterns if he is allowed to work on horizontally partitioned data. Although the parties realize that combining their data has some mutual benefit, none of them is willing to reveal its own database to other parties. Privacy preservation can mean many things, for example, protecting specific sensitive values of individuals, hiding the link between attribute values and the individuals they are applied to, and protecting the sources. The proposed technique allows each site to sponsor the required data to the third party without modifying the structure of the data. The third party can apply mining techniques or algorithms, without modification, on the joined dataset to get results as accurate as if mined from the actual dataset. At the same time, the data miner or any adversary can not understand (interpret) the results or patterns. Also, any authorized researcher who works on this data set can only interpret the results with the help of the data owners. Thus the individuals' privacy is preserved.

2 RELATED WORKS

Privacy preserving data publishing for the benefit of researchers tends to study different transformation methods associated with privacy. These techniques include methods such as randomization, k-anonymity, and l-diversity. The randomization method is a technique that adds noise to the original data. Only the perturbed values and the distribution function used for perturbation are given for analysis. However, it is impossible to reconstruct the exact distribution of the original data, and the accuracy level in estimating the data distribution is sensitive to the reconstruction algorithm. This is proved by Agrawal and Aggarwal (2001). The micro data records contain the actual unperturbed data associated with the individuals to support the development of new data mining algorithms while satisfying legal requirements. Micro data records may contain identifying attributes, sensitive attributes, quasi identifier attributes, and neutral attributes. Identifying attributes, such as *name* and s*ocial security number*, are not released to protect privacy. Some attributes, such as *disease* and *income,* are sensitive attributes whose actual values should not be published without the individual's consent. There may be other attributes, such as *age, sex*, and *zip-code*, that in combination are called quasi identifiers. These quasi identifying attributes may be used to recover personal identities (the linking attack problem). Personal identity leads to sensitive attribute disclosure. Micro data may also contain neutral attributes such as *length_stay_hospital.* Micro data are allowed to be released only when the individuals are unidentifiable.

2.1 Anonymization

K-anonymization from Samarati (2001) is the first anonymization principle used in the literature to counter the linking attack problem. It requires each quasi identifying (QI) group to contain at least K tuples. All the values of the tuples in the QI group for these set of attributes (QI attributes) are made the same by transformation procedures such as suppression, generalization, etc. Because K-anonymity does not place any constraint on sensitive attributes, there exists the "homogenous" QI-group, where all tuples possess exactly the same sensitive attribute value. This homogeneity offers virtually no protection against linking attacks. K-anonymity only prevents association between individuals and tuples instead of association between individuals and their sensitive values. Motivated by this problem, the L-diversity principle (Machanavajjhala et al., 2006) was developed. It demands at least L *well-represented* sensitive attribute values in every QI-group, and it discusses only categorically sensitive data. None of these existing anonymization principles can effectively prevent *proximity* or *divergence breaches*.

A *Proximity breach* is a privacy threat specific to numerically sensitive attributes in anonymized data publication. Such breaches occur when an adversary concludes with high confidence that the sensitive value of a victim individual must fall in a short interval—even though the adversary may have low confidence about the victim's actual value. For example, two persons X and Y having the same Quasi identifying values (generalization of *DOB*, *gender*, *zip code*) have incomes of 10,000 and 10,050 respectively. Assume an adversary who can guess the QI group of X and is interested in knowing the income of X from the 2-anonymized data table. Even though the adversary guesses the actual income with 50% probability, he gets a short range of 10,000 to 10050 with 100% confidence. This short value range is almost similar to the disclosure of the actual value of 10,000. This problem is handled by Li, Tao, and Xiao (2008) who introduced a novel principle called (e,L)-anonymity. Intuitively, this principle demands that, given a QI-group G, for every sensitive numerical value x in G, at most 1/L of the tuples in G can have sensitive values "similar" to x, where the similarity is controlled by e. However, this method concentrates on micro data that contain only a single numerically sensitive attribute. The proposed simple categorical grading based transformation method avoids proximity breach and also can be applied to any number of numerical attributes in the data table.

We have observed a new privacy threat in a K-anonymized data set and called it a divergence breach. A *divergence breach* occurs in categorically sensitive attributes when an adversary concludes, with high confidence, a completely irrelevant value for the sensitive attribute of the victim, even though the adversary may have low confidence about the victim's actual value. This is a more dangerous situation than allowing the adversary to know the actual value of the victim. For example, two patients X and Y having the same Quasi identifying values (generalization of *DOB*, *gender*, z*ip code*) have the diseases gastritis and stomach cancer, respectively. Assume an adversary who can guess the QI group of X and is interested in knowing the disease of X from the 2-anonymized data table. He guesses the actual disease with 50% probability. At the same time the adversary links person X with stomach cancer with a 50% probability, which person X may not like. Because of the anonymization performed, X and Y are in the same QI group, and X is linked with cancer. However, X may even be willing to disclose his actual disease to avoid linking him with the totally irrelevant disease, cancer. Even though the researcher is not bothered by the adversary getting wrong information about individuals X and Y, the individuals may be bothered that they are linked with totally irrelevant information. Therefore, anonymization leads to a divergence breach. In the proposed method, the mapping table based transformation is used for categorically sensitive attributes. The transformed table prepared from the original table is given alone for an analysis that limits both the proximity and divergence breaches.

2.2 Distributed Database

A database on which the mining is to be performed may be horizontally or vertically partitioned. A simple approach to privacy preserving data mining over multiple sources that are not willing to share data is to apply existing techniques and tools at each site independently and combine the results. This, however, will not give globally valid results because of duplicated data at different sites. Also it is not possible to detect cross site correlations. Another approach is to perturb the local data (by adding "noise") before the data mining process and mitigate the impact of the noise from the data mining results by using reconstruction techniques (Agrawal & Srikant, 2000). However, it is impossible to reconstruct the original data set, and also the accuracy depends on the reconstruction algorithm (Agrawal & Aggarwal, 2001).

The problem of distributed privacy preserving data mining overlaps closely with cryptography in determining secure multi-party computations. Many of these techniques work by sending changed or encrypted versions of the inputs to one another in order to compute the function with different alternative versions followed by an oblivious transfer protocol to retrieve the correct value of the final output. The algorithms for secure multiparty computation over horizontally partitioned data sets include the Naïve Bayes classifier (Kantarcioglu & Vaidya 2004), the Support Vector Machine (SVM) classifier with non linear kernels (Yu, Jiang, & Vaidya, 2006), Clustering (Inan, Saygin, Savas, HIntoglu & Levi, 2007; Jagannathan & Wright, 2005; Jagannathan, Pillaipakkamnatt & Wright, 2006), and Association Rule Mining (Kantarcioglu & Clifton 2004). Kantarcioglu and Clifton (2004) incorporated cryptographic techniques to minimize the information shared while adding little overhead to the Association Rule mining task. However, the proposed method aims at the privacy preserving data mining and thus enables most or all data mining methods to be applied without restriction.

3 PRIVACY PRESERVING MODEL

The proposed framework allows us to systematically transform normal data mining computations to secure multi-party computations. The problem is defined as this: There are 'n' parties, each having a private database, who want to jointly conduct a data mining operation on the union of their databases. How can these parties accomplish this without disclosing their database to the other parties or a third party? The framework of our privacy preserving mining model is shown in Figure 1.

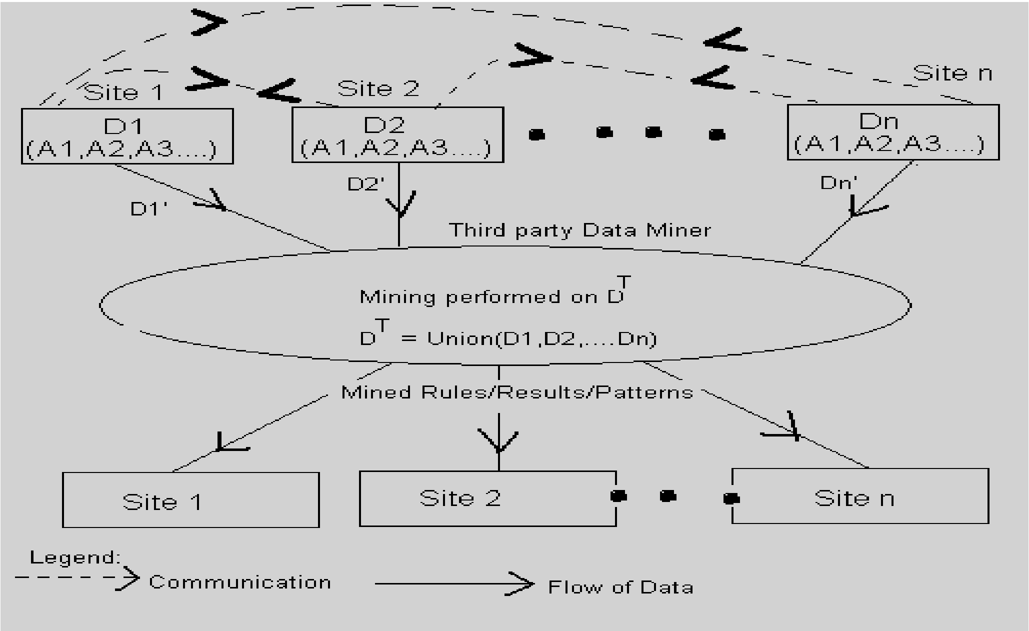


Figure 1. Framework of the Privacy Preserving Model

3.1 Data Flow

All the n parties available in Site_1 to Site_n have their own datasets D_l to D_n, each having the same set of attributes. In some situations, only a part of the data set needs to be kept confidential. These attributes are sensitive attributes, and none of the other attributes need treatment. All the parties want to jointly conduct data mining operations on a single database D, which is formed by the union of all the datasets $D_{1,}$ D_2… and D_n to get better results. However, to preserve the privacy of the individual databases, the third party data miner is allowed to work with a single database D^T, which is formed by the union of all the transformed data bases D_l', D_2'…D_n', where D_l'= {A_{1T}, A_{2T}…}, D_2'={A_{1T}, A_{2T}…}, and D_n'={A_{1T},A_{2T}…}. A_{xT} is the transformed value of the sensitive attribute A_x. An attribute is called *sensitive* if the individual is not willing to disclose it or if an adversary must not be allowed to discover the value of that attribute.

The method of converting the attribute A_x to A_{xT} is explained in Section 3.2. Each site communicates with the other sites to discuss the transformation parameters and supply the transformed data bases D_l', D_2'…D_n' to the third party data miner. Data miners who work on the transformed single data base D^T can perform any data mining task as if they are working on the original data. However, they cannot interpret the results\rules\patterns. They can declare the results to all parties that have participated in the data sharing. The individual parties containing the transformed attribute values of their own data bases can interpret the results. Because only the actual value of the results\rules\patterns is known by all parties, the individual's privacy is preserved.

3.2 Transformation Method

The transformation of attribute A_x to A_{xT} is based on the data type of attribute A_x. The graded grouping technique is used for the numerical data type, and a mapping table is used for the categorical data type. The graded grouping technique is a transformation method that maintains the correlation factor of nearly 1 between the transformed values and the original values. This graded grouping approach applied to the numerical attributes is shown in Figure 2.

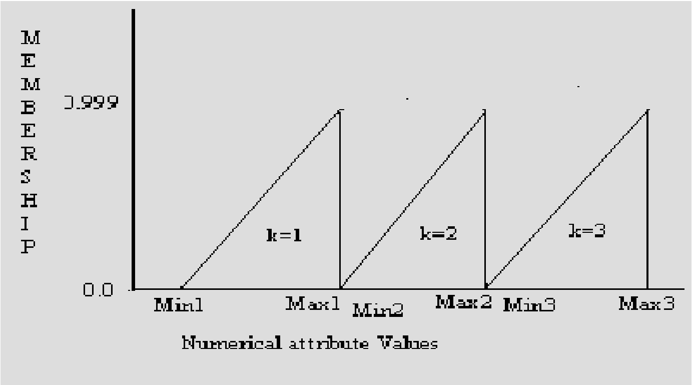


Figure 2. Graded Grouping Membership for Categories k=3

To convert the actual values of a single numeric attribute value, the following steps are followed: The first step is to fix the number of categories (k) for the numeric data domain. The second step is, for each category $C_1 \ldots C_k$, the to fix the maximum and minimum value in such a way that a non overlapping continuous range results. Here the range (maximum-minimum) for each category may or may not be uniform. If the uniform range is considered for each category, then the correlation factor between the original and transformed values is one. Otherwise, it will decrease but will maintain a positive linearity. The third step is to fix the category (Ci) for each actual value(x) to which it belongs and find the membership value m(x) using Eq. (1)

$$\begin{aligned} m(x) &= 0.0 && \text{if } x = \min(Ci) \\ &= (x - \min(C_i)) / (\max(C_i) - \min(C_i)) && \text{if } \min(C_i) > x > \max(C_i) \\ &= 0.999 && \text{if } x = \max(C_i) \end{aligned} \tag{1}$$

The fourth step is to replace the actual value with a new transformed value n(x), which can be calculated by adding the membership value m(x) with the value of that category V_i. The four steps are summarized and derived as a simple formula for transformation and shown as Eq. (2) in Section 3.2.1.

3.2.1 Algorithm for transformation

1. Let x_j be the actual value to be transformed where $1 \le j \le m$
2. Let k be the number of categories or partitions or groups
3. Let V_i be the value or name of the ith partition with $V_i < V_{i+1} < V_{i+2}$, and $V_i = 2q$ or $2q+1$ where q=0,1,2…
4. Let R_i be the range of the ith partition where $R_i = U_i - L_i$ and $U_i < L_{i+1}$ and where U_i is the upper limit of ith partition and L_i is the lower limit of ith partition
5. Let $\mathbf{y_j}$ be the publishable value of x_j
6. Then for$1 \le j \le m$ and 1<=i<=n

$$\begin{aligned} \mathbf{y_j} &= V_i + (x_j - L_i)/(U_i - L_i) && \text{for } L_i < x_j < U_i \\ &= V_i && \text{for } L_i = x_j \\ &= Vi + 0.999 && \text{for } U_i = x_j \end{aligned} \tag{2}$$

The mapping technique for categorical values is nothing but a transformation of the actual values to alias names. A sample mapping table for the attribute *disease* is shown in Table 1. The mapping table is preserved by the individual sites. Because alias names are given randomly, the actual values cannot be guessed without the help of a mapping table. Thus, a third party data miner can not interpret the rules/results/patterns that they have mined.

Table 1. A Sample Mapping Table

Original Value	Transformed to
Gastritis	Illness_1
Flu	Illness_2
Stomach Cancer	Illness_3
Throat infection	Illness_4
Mouth ulcer	Illness_5

3.3 Experimental Setup

The data miner's first job is to perform a union operation on the various transformed attributes A_{1T}, A_{2T}… and the other non sensitive attributes to form a single table D^T. This table D^T can be used for any data mining task. It was decided to conduct the experiment on a real data set, the adult database from the UCI machine learning repository (Newman, Hettich, Blake, & Merz, 1998) with 35,561 records. The attributes *age, work class, education, marital status, occupation, relation, race, sex*, and *country* were considered for analysis, assuming that different records for all the attributes are received from different sites. The attribute *age* was considered as a sensitive numerical attribute and Age^T was calculated by the transformation algorithm. Similarly, *education* was considered as a sensitive categorical attribute, and hence $Education^T$ was formed from a mapping table. The algorithm was implemented in Java standard Edition 5.0 and made to run on the Intel® Core2 Duo, 1.8 GHz, 1GB RAM system, which took only 28sec for generating the privacy preserving adult data set D^T. The various data mining tasks were performed on both tables, the original table D and the new table D^T, using the tool WEKA (Witten & Frank, 2005), and the results were compared.

4 EXPERIMENTAL RESULTS ANALYSIS

Mining tasks such as classification, clustering, and Association Rule mining were performed on both the original (D) and transformed (D^T) tables. The resultant rules\patterns were analyzed for accuracy (information) and privacy. Any privacy preserving data mining technique can be evaluated with parameters such as data utility, information loss, and the resistance accomplished by privacy algorithms.

4.1 Data Utility

Utility is the usefulness of the data in arriving at better and more accurate results, especially when mining is performed. Therefore, any transformation done on the data (to protect privacy) needs to maintain the utility of the data. Some forms of transformation are suppression and generalization. The suppression technique results in missing data. Missing values cannot be handled by many mining tools, and thus the utility of the data is reduced. Because the proposed transformation techniques preserve the structure and data type, the utility of the data before and after transformation remains the same.

4.2 Information Loss

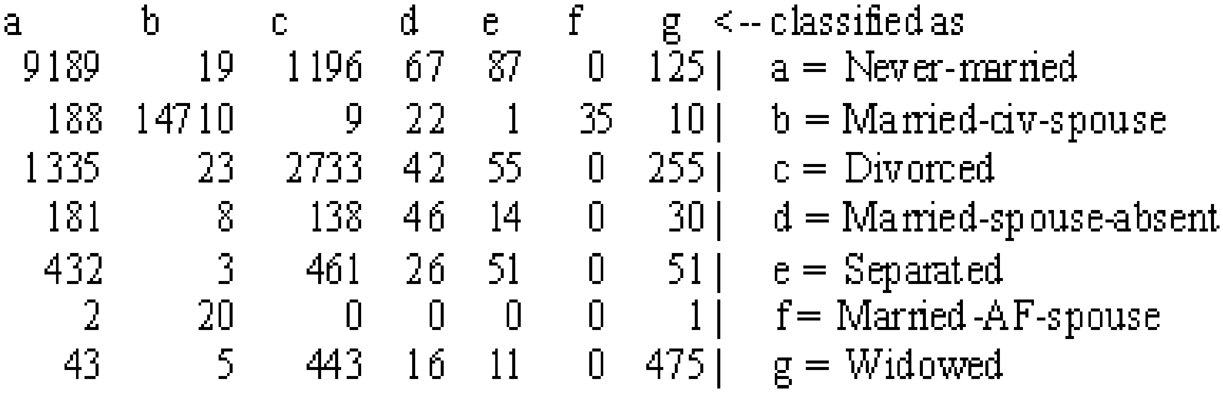
Both proximity and divergence breaches exist as long as a QI group exists. Therefore, in the data base given for mining, no QI group is formed. Because there is no generalization to form a QI group, there is no information loss. The information loss measurement of any data table depends on the specific data mining task performed. For example, in the case of association rule mining, information loss can be measured by counting the number of rules framed for the given support and confidence, before and after transformation. From Table 2, we may conclude that information loss is nil because the same number of rules were framed from both the original and the transformed table, irrespective of the association rule mining algorithm used. Because Weka cannot handle numerical data for association rule mining, the attribute *age* was omitted for this task.

Table 2. Comparison of Association Rules Formed

	Original & Transformed Table	
Scheme	**Number of Rules**	**Confidence**
Apriori	10	>=0.91
Tertius	11	>=0.95
Predictive Apriori	100	>=0.9941

Consider another example: K-anonymization using generalization is a privacy preserving algorithm developed earlier. The efficiency of any data mining algorithm increases by increasing K. The information loss can be measured by classification accuracy. Increasing the value of K may even improve the results (LeFevre, DeWitt, & Ramakrishnan, 2006; Fung, Wang, & Yu, 2007). The reason for this is that generalization procedures form a QI-group based on certain similarities, and this makes the classification task easier. However, by using generalization, the minute information is lost, and hence increasing K increases information loss. However, in the proposed method, there is no hidden information loss.

The classification done on original and transformed data tables proves that the algorithm does not affect classification accuracy. Classification accuracy in adult data is 85.55%, with marital status as a classification variable. The same accuracy is obtained from both the original and the transformed tables. The resultant confusion matrix is shown in Figure 3. The density based clustering task performed using Weka on both the original and transformed tables resulted in the sum of squared errors within clusters to be 92466.



```
a       b      c     d    e    f     g  <-- classified as
 9189     19   1196   67   87    0   125 |  a = Never-married
  188  14710      9   22    1   35    10 |  b = Married-civ-spouse
 1335     23   2733   42   55    0   255 |  c = Divorced
  181      8    138   46   14    0    30 |  d = Married-spouse-absent
  432      3    461   26   51    0    51 |  e = Separated
    2     20      0    0    0    0     1 |  f= Married-AF-spouse
   43      5    443   16   11    0   475 |  g = Widowed
```

Figure 3. Resultant Confusion Matrix

4.3 Resistance Accomplished by the Algorithms

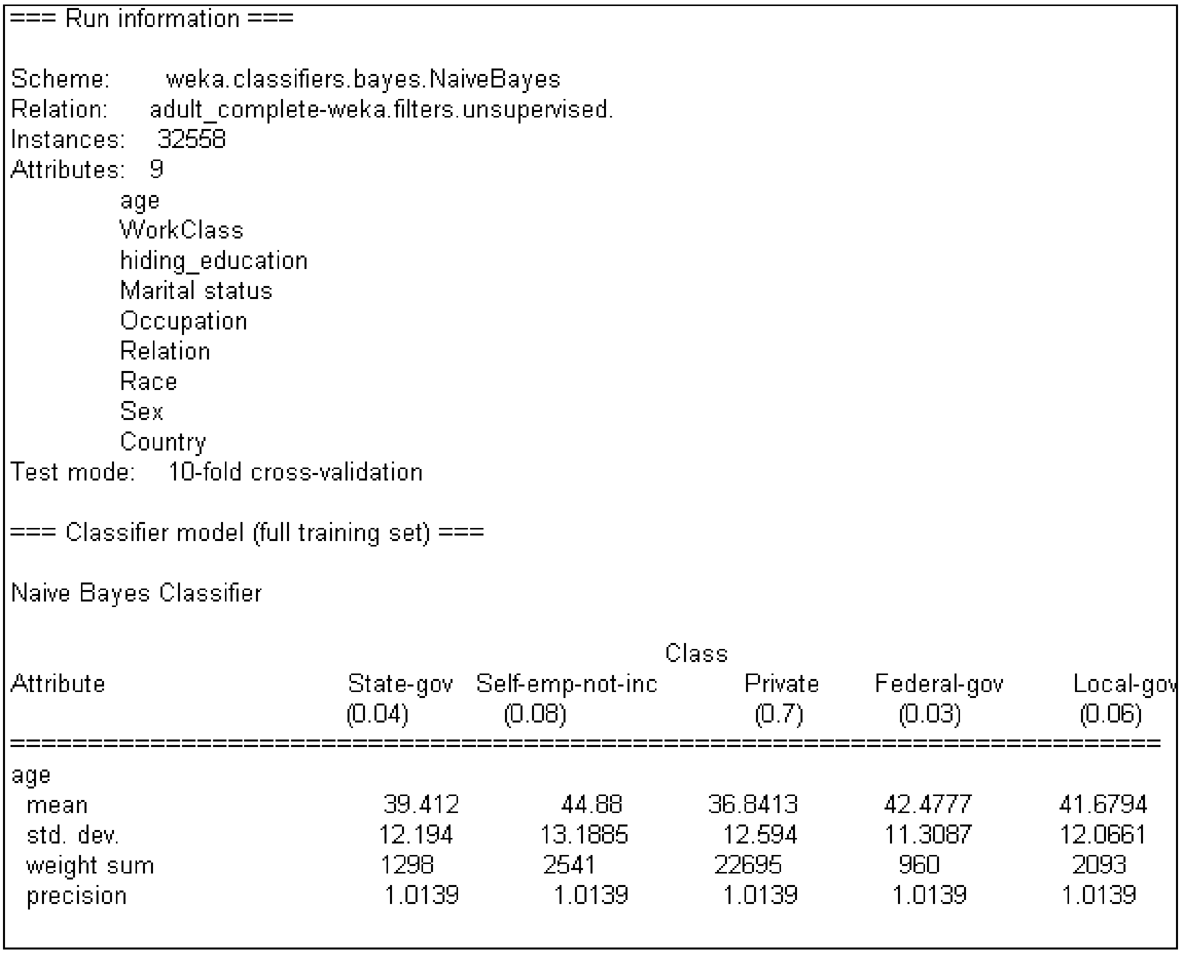
Privacy preserving techniques adopted in the literature use different transformation methods for different data mining tasks. However, if the data miner performs some other task, the expected privacy protection may not be provided by the specific algorithm. On the other hand, if the proposed transformed table is given for mining, privacy is preserved. Various data mining tasks performed on both the original and transformed table yield the same set of rules/ results, which cannot be interpreted by the data miner. If an authorized researcher performs the mining task, he can interpret the results with the help of the data owner.

For example, actual values and transformed values are shown in Table 3, where k is the number of categories assumed. The transformed value 1.667 represents an actual value of 25 when k=3, but the actual value is 30 when k=4. Also, the actual value 90 is represented by 5.99, 4.99, or 3.99 depending on the value of K. Therefore, the actual value can not be guessed accurately, thereby privacy of the numerical sensitive attribute is preserved.

Table 3. A sample transformation for numerical attribute

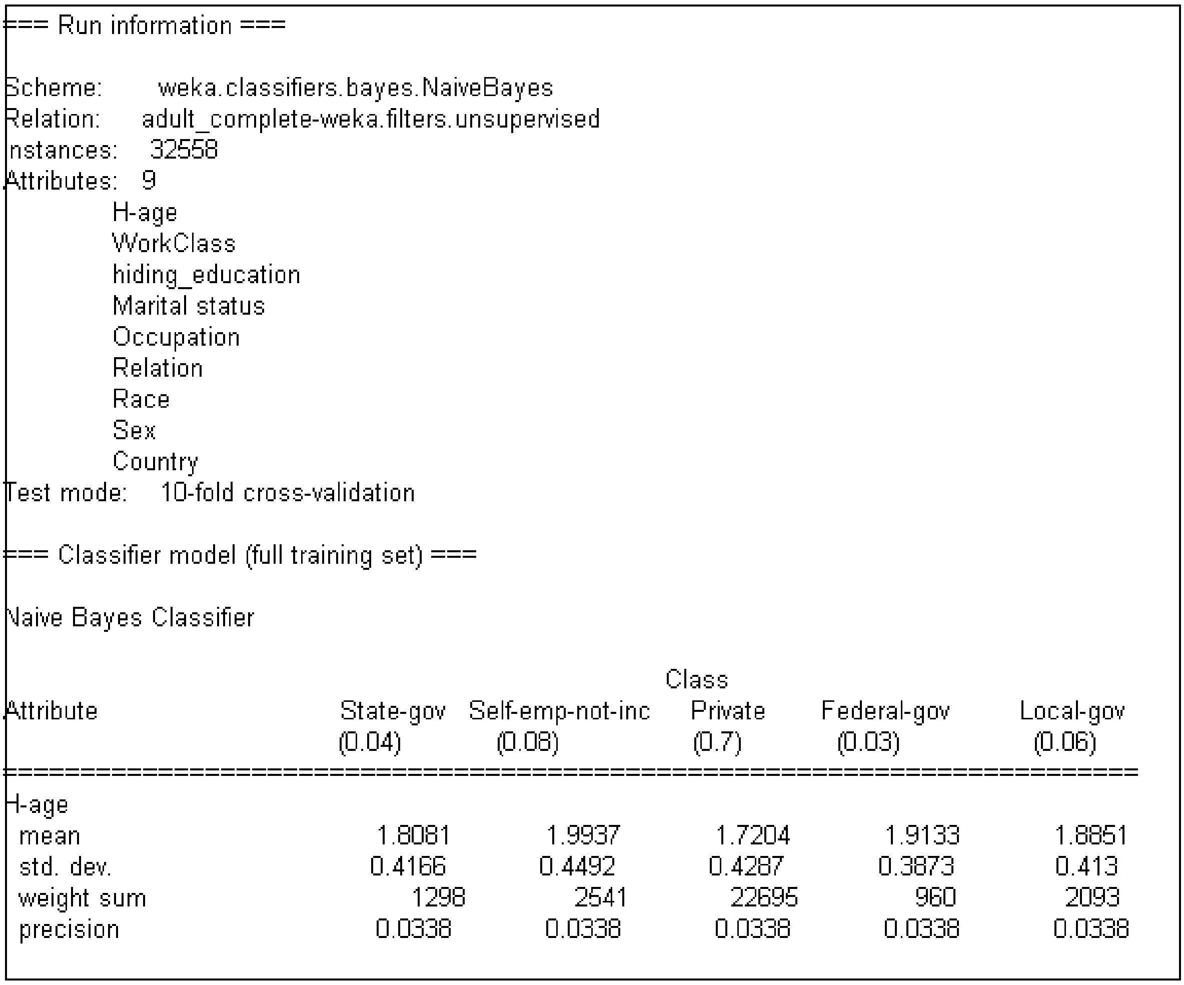
	Transformed Values when		
Actual Value	k=5	k=3	k=4
30	2	1.667	1.8
40	2.667	2	2.267
70	4.667	3.16	3.80
25	1.667	1.5	1.6
15	1	1.167	1.2
58	3.867	2.68	3.28
73	4.867	3.88	3.92
37	2.467	1.9	2.13
90	5.99	3.99	4.99
Correlation Coefficient	1	0.983	0.982

Figures 4 and 5 show the output snapshots of WEKA when the Naïve Bayes classification (with work_class as the class variable) is done on the original table (with *age* as one of the attributes) and the transformed table (with 'H-Age' attribute), respectively. Except for the average sum, all the other parameters such as Mean, Standard deviation, and precision of the attribute *age* are different in the original and transformed tables. This shows that even the statistical parameters cannot leak the actual information.



```
=== Run information ===

Scheme:       weka.classifiers.bayes.NaiveBayes
Relation:     adult_complete-weka.filters.unsupervised.
Instances:    32558
Attributes:   9
              age
              WorkClass
              hiding_education
              Marital status
              Occupation
              Relation
              Race
              Sex
              Country
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

Naive Bayes Classifier

                                                    Class
Attribute                   State-gov  Self-emp-not-inc      Private    Federal-gov     Local-gov
                               (0.04)            (0.08)        (0.7)         (0.03)        (0.06)
==================================================================================================
age
  mean                         39.412             44.88      36.8413        42.4777       41.6794
  std. dev.                    12.194           13.1885       12.594        11.3087       12.0661
  weight sum                   1298              2541         22695            960           2093
  precision                    1.0139            1.0139       1.0139         1.0139        1.0139
```

Figure 4. A portion of the Snapshot of the Naïve Bayes Classification of the original 'Adult Data'



```
=== Run information ===

Scheme:       weka.classifiers.bayes.NaiveBayes
Relation:     adult_complete-weka.filters.unsupervised
nstances:     32558
Attributes:   9
              H-age
              WorkClass
              hiding_education
              Marital status
              Occupation
              Relation
              Race
              Sex
              Country
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

Naive Bayes Classifier

                                                    Class
Attribute                  State-gov  Self-emp-not-inc  Private   Federal-gov     Local-gov
                              (0.04)          (0.08)      (0.7)        (0.03)        (0.06)
==========================================================================================
H-age
 mean                         1.8081          1.9937     1.7204        1.9133        1.8851
 std. dev.                    0.4166          0.4492     0.4287        0.3873         0.413
 weight sum                     1298            2541      22695           960          2093
 precision                    0.0338          0.0338     0.0338        0.0338        0.0338
```

Figure 5. A portion of the Snapshot of the Naïve Bayes Classification of the transformed 'Adult' data

Box plots drawn with the actual values (first column of Table 3) and transformed values (second column of Table 3, when k=5) are shown in Figures 6 and 7, respectively. The five point summary of actual values is Q1=15, Q2=27.5, Q3=40, Q4=71.5, and Q5=90. The five point summary of transformed values is Q1=1, Q2=1.8335, Q3=2.667, Q4=4.767, and Q5=5.99. It is interesting to note that the shapes of both plots are exactly the same, indicating that the properties of the values or the spread of the numerical values is maintained even after transformation.

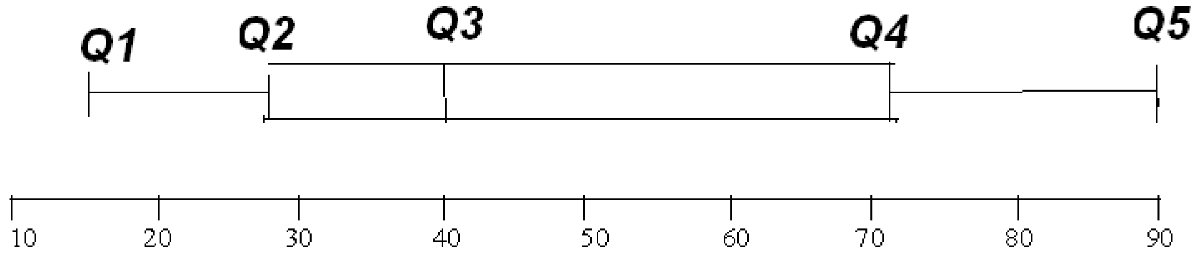


Figure 6. Box plot for the actual data set in Table 3

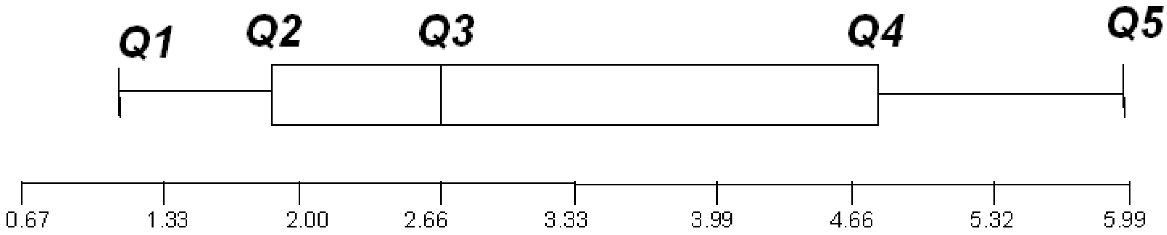


Figure 7. Box plot for the transformed values in Table 3 when k=5

5 CONCLUSION

Numerous techniques, such as K-anonymity, L-diversity, and Randomization, that preserve an individual's privacy during data mining, exist in the literature. However, there always exists a trade off between privacy and information. When an algorithm improves the privacy level, information loss is also increased. Most data publishing methods attempt to change (generalize) an individual's information by merging it with that of a group so that the individual's information becomes unidentifiable. However, in this kind of generalization, actual minute information of the individual is *lost*, which in turn affects the mining results. The proposed method does not lose any minute information and at the same time does not disclose actual values.

Furthermore, to improve the efficiency of privacy preserving algorithms, work load aware algorithms were developed by LeFevre, DeWitt, and Ramakrishnan (2006). However, these task dependent algorithms are good enough to do the job (of preserving privacy) only when the data miner is trusted. Otherwise, these algorithms can not preserve privacy. The proposed algorithm, however, is a task independent privacy preserving technique.

In the proposed approach, horizontally partitioned data sets are combined to form a single centralized data set upon which mining is allowed to be performed. This reduces the complexity of the algorithm while preserving privacy, information, and utility. Finally, any number of sensitive attributes of numerical or categorical data types can be handled in this approach.

6 REFERENCES

Adam, N. & Wortmann, J. C. (1989) Security-Control Methods for Statistical Databases: A Comparison Study. *ACM Computing Surveys*, 21(4), pp. 515-556.

Aggarwal, C. C. (2005) On k-anonymity and the curse of dimensionality. *VLDB Conference*, pp. 901-909, Trondheim, Norway.

Aggarwal, G., Feder, T., Kenthapadi, K., Khuller, S., Motwani, R., Panigrahy, R., Thomas, D., & Zhu, A. (2006) Achieving Anonymity via Clustering. *ACM SIGMOD Symposium on PODS,* Chicago, USA, pp. 153-162.

Agrawal, D. & Aggarwal, C. C. (2001) On the Design and Quantification of Privacy- Preserving Data Mining Algorithms. *ACM SIGMOD Symposium on PODS*, pp. 247-255.

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

Fung, B. C. M., Wang, K., & Yu, P. S. (2007) Anonymizing classification data for privacy preservation. *IEEE Transactions on Knowledge and Data Engineering*, 19(5), pp. 711-725.

Inan, A., Saygin, Y., Savas, E., HIntoglu, A., & Levi, A. (2007) Privacy Preserving Clustering on Horizontally Partitioned Data. *Data and Knowledge Engineering, 63(3),* pp. 646-666.

Jagannathan, G., Pillaipakkamnatt, K., & Wright, R. (2006) A New Privacy Preserving Distributed K- clustering Algorithm. *SIAM Conference on Data Mining*, Bethesda, Maryland, pp. 494-498.

Jagannathan, G. & Wrigh,t R. (2005) Privacy Preserving Distributed K-means clustering over arbitrarily partitioned data. *ACM SIGKDD ,* Chicago, USA, pp. 593-599.

Kantarcioglu, M. & Clifton, C. (2004) Privacy Preserving Distributed Mining of Association Rules on Horizontally Partitioned data. *IEEE Transactions on Knowledge and Data Engineering*, 16(9), pp.1026-1037.

Kantarcioglu, M. & Vaidya, J. (2004) Privacy Preserving Naïve Bayes Classifier for Horizontally Partitioned Data, *Workshop on Privacy, Security, and Data Mining,* Melbourne.

LeFevre, K., DeWitt, D. J., & Ramakrishnan, R. (2006) Workload-aware anonymization. *12th ACM SIGKDD*, New York, USA, pp. 277-286.

Li, J., Tao, Y., & Xiao, X. (2008) Preservation of Proximity Privacy in Publishing Numerical Sensitive Data. *ACM SIGMOD*, Canada, pp. 473-486.

Machanavajjhala, A., Gehrke, J., Kifer, D., & Venkitasubramaniam, M. (2006) l-Diversity: Privacy Beyond k-Anonymity. *ACM Transactions on Knowledge Discovery from Data, 1(1).*

Newman, D. J., Hettich, S., Blake, C. L. & Merz, C. J. (1998) UCI Repository of Machine Learning Databases. Retrieved June 7, 2010, from the World Wide Web: http://archive.ics.uci.edu/ml/

Samarati, P. (2001) Protecting respondents' identities in micro data release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6), pp. 1010-1027.

Sweeney, L. (2002) Achieving k-anonymity privacy protection using generalization and suppression. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems.* 10 (5), pp. 571-588.

Wenliang, Du & Zhijun, Zhan (2003) Using Randomized Response Techniques for Privacy-Preserving Data Mining. *ACM SIGKDD*, Washington D.C.*,* pp.505-510.

Witten, I. H. & Frank, E. (2005) *Data Mining: Practical machine learning tools and techniques. 2nd Edition*. Morgan Kaufmann: San Francisco.

Yu, H., Jiang, X. & Vaidya, J. (2006) Privacy Preserving SVM using nonlinear kernels on horizontally partitioned Data. *Symposium on Applied Computing*, pp. 603-610.

(Article history: Received 26 October 2009, Accepted 14 May 2010, Available online 23 June 2010)