A fuzzy programming approach for data reduction and privacy in distance-based mining

Shibnath Mukherjee, Zhiyuan Chen* and Aryya Gangopadhyay

Department of Information Systems
University of Maryland, Baltimore County (UMBC)
1000 Hilltop Circle
Baltimore, MD 21250, USA
Fax: +1–410–455–1073
E-mail: shibm1@umbc.edu
E-mail: zhchen@umbc.edu
E-mail: gangopad@umbc.edu
*Corresponding author

Abstract: With the explosive growth of data and its distributed sources, there are increasing needs for secure cooperative data analysis. The issue of data reduction to decrease communication overheads and the issue of preservation of privacy of the shared data are becoming important. However, existing privacy preserving techniques do not work well for distance-based mining because they do not preserve distances. Besides, most of them either do not reduce data or are tied to very specific mining algorithms. Using the unitarity and energy compaction property of Fourier transforms, this paper proposes a novel framework to preserve privacy and reduce data size, yet preserve Euclidian distances. A fuzzy programming approach for selection of Fourier coefficients is proposed to optimise the objective of preserving Euclidean distances and obtaining privacy and data reduction through coefficient suppression. Experiments demonstrate the superiority of the proposed approach over the existing ones.

Keywords: privacy; data mining; fuzzy programming; distance-based learning; information and computer security; data reduction.

Reference to this paper should be made as follows: Mukherjee, S., Chen, Z. and Gangopadhyay, A. (2008) 'A fuzzy programming approach for data reduction and privacy in distance-based mining', *Int. J. Information and Computer Security*, Vol. 2, No. 1, pp.27–47.

Biographical notes: Shibnath Mukherjee was a PhD student in the Department of Information Systems, University of Maryland Baltimore County when this work was conducted. He currently holds a position at IBM India. He completed his Bachelor's Degree in Electronics and Telecommunication Engineering and MBA in Information Systems and Economics. His areas of interest and research include mathematical transforms for privacy preserving distributed data mining and data mining over mobile networks.

Zhiyuan Chen is an Assistant Professor at Information Systems Department, UMBC. His research interests include privacy preserving data mining, data navigation and visualisation, XML, automatic database tuning and database compression.

Copyright © 2008 Inderscience Enterprises Ltd.

Aryya Gangopadhyay is an Associate Professor and the Graduate Programme Director of the Department of Information Systems at UMBC. He has a PhD in Information Systems from Rutgers University. Dr. Gangopadhyay's current research interests are in the areas of data mining and warehousing. Dr. Gangopadhyay has published over 60 peer-reviewed articles in journals, conference proceedings, and book chapters, and three books.

1 Introduction

With tremendous increase in volume of data over time, most of the businesses keep looking for solutions of three issues. The first one is to find a cheaper way to ship data for analysis, which means reducing communication overheads in data transfer, as more and more businesses are outsourcing their data analysis process. The second one is the issue of secure exchange of data for cooperative analysis. In the process, privacy of individual data is a big concern. The third one is of establishing one unified transform framework that will achieve both data reduction and privacy and accommodate a range of popular mining algorithms (sharing some commonalities). Quite a few approaches such as random perturbation/projection approaches (Agrawal and Srikant, 2000; Liu *et al*., 2006) have been proposed to solve these issues. However as discussed in the related work section, the perturbation methods suffer from inaccuracy in preserving Euclidean distances and do not reduce data. The projection methods do reduce data but suffer from inaccuracy though to a lesser degree than element-wise additive perturbations. The secure multiparty computations (Du and Atallah, 2001), also discussed in the next section, extend the process of secured analysis for distributed data sources, but have a serious drawback. Most of the existing approaches are tied to very specific algorithms and are not generalisable even to a class of similar algorithms. This drawback leads to a separate privacy method for each mining algorithm.

This paper focuses on building a unified transform framework to preserve privacy and to reduce data volume while allowing the whole range of distance-based mining algorithms to work accurately on the modified data. The algorithms presented in this paper are based on two useful properties of discrete Fourier-related transforms. First is that, they are unitary transforms (thus Euclidean distances between data vectors are preserved in transformed domain). Second is their ability to compact energy of correlated data. These properties have been much used in the field of signal processing and image processing (Oppenheim and Schafer, 1999). They have also been used in Agrawal *et al*. (1993) for similarity search in sequence databases and in Egecioglu *et al*. (2004) for approximating inner products. The algorithms proposed in this paper treat each data row as a discrete sequence and apply Discrete Cosine Transform (DCT is selected because it returns real numbers, although other Fourier-related transforms will work as well). For most of the real life datasets, energy is concentrated on a small set of transformed coefficients common across the majority of rows. Thus selecting such coefficients will optimise the trade-off between preserving distance and provide privacy and data reduction through coefficient suppression. However, the major challenge is to select such coefficients. One could certainly select a fixed number of coefficients with the highest average energy. However, it is difficult to decide the number of coefficients to be selected because too large a number will lead to poor data privacy and data reduction and

too small a number will lead to a big loss of Euclidean distance. A better solution is to select the minimal set of coefficients which retain at least a certain percentage of energy across all rows. This can be modelled as an integer linear programming problem. However it is extremely difficult to select the retention value correctly. If chosen improperly, the solution will have all coefficients selected, resulting in no data reduction. Thus it becomes necessary to relax the specification of the percentage retention rate for each of the rows. Further, the problem becomes extremely complicated with increasing number of rows because there is one constraint per row. Both problems can be solved by incorporating a simple fuzzy linear programming approach used previously in many instances of uncertain parameter optimisation problems like multi-criteria capital budgeting (Kahraman and Ulukan, 1999) and energy planning (Canz, 1999). All these problems inherit the principle developed by Bellman and Zadeh (1970) and later refined and extended by Zimmermann (1978). It will be shown that posing the problem as a fuzzy optimisation increases the flexibility of the approach to a huge extent and makes it suited to handle massive datasets with relatively nominal calculation overhead.

The contributions of this paper are summarised as follows:

- This paper proposes a fuzzy programming approach to select optimal coefficients to prune. It then proposes three algorithms for centralised, horizontally partitioned, and vertically partitioned data.
- The paper conducts extensive experimental evaluation to compare the proposed methods with existing methods for two popular Euclidean-distance-based mining algorithms: K-means clustering and K-nearest neighbour classification. The results demonstrate the superiority of the proposed methods.

The paper is organised as follows. Section 2 discusses related work. Section 3 describes proposed algorithms. Section 4 reports the results from extensive experiments. Section 5 concludes the paper.

2 Related work

Existing work on privacy preserving mining often uses element-wise random perturbation approaches that add or multiply random noise to each data element such that individual data values are distorted while the underlying distribution can be reconstructed with fair degree of accuracy (Agrawal and Srikant, 2000). With such an approach, Euclidean distances between individual data points are not preserved. Thus, many widely used distance-based mining algorithms such as K-means clustering (Duda and Hart, 1973) and the K-nearest neighbour classification (Cover, 1968; Duda and Hart, 1973) perform poorly on additively perturbed data. Further, element-wise techniques perturbation do not reduce data size and they have a serious privacy breach as pointed in Kargupta *et al*. (2003) through application of simple spectral filtering techniques to recover original data values.

The example in Figure 1 shows two randomly generated clusters of data with two attributes following two 2-D Normal distributions. Figure 2 shows the same data but added with a random noise for each attribute following normal distribution with mean equals zero and standard deviation equals 0.25. Clearly, the Euclidean distance is not preserved in the perturbed data, and the two clusters in Figure 1 no longer exist in Figure 2.

Figure 1 Original data with two clusters

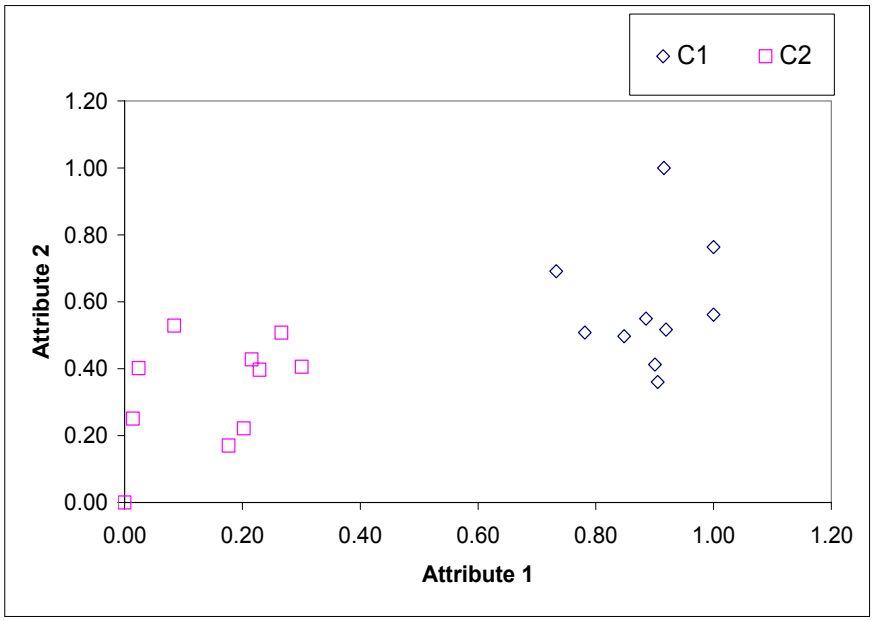
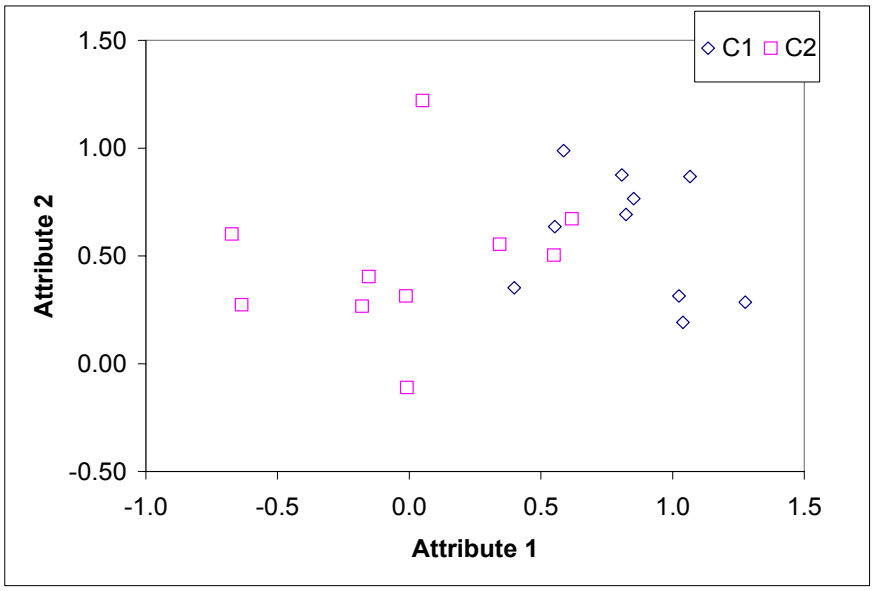


Figure 2 Perturbed data



Random projection methods have been proposed to address the above shortcomings (Liu *et al*., 2006; Oliveira and Zaïane, 2003). These methods are based on Johnson and Lindenstrauss (1984) lemma which places bounds on Euclidean distance distortion due to any dimensionality reduction transform. Let $X_{n\times m}$ be a dataset, $R_{m\times k}$ be a matrix generated with entries randomly chosen from a given distribution with zero mean and normalised to unit length across columns. Now if $k < m$, X multiplied by R results in a reduced dataset whose expected inner products are equal to the inner products of the original data. However, the random projection methods suffer from the loss of Euclidean distances due to non-orthogonality of the matrix R. Note that Euclidean distances will be kept only if R is strictly orthogonal, and orthogonal matrices are always square ones. However, to achieve data reduction, k must be less than m, thus R cannot be orthogonal. The fact that R is orthogonal in expected sense (shown by the authors) does not guarantee that Euclidean distances will be preserved accurately. It will be shown in the experiments (see Section 4) that random project methods lead to poor mining quality.

There also exists work on secure multi party computations (SMC). Please refer to Du and Atallah (2001) for a comprehensive review. However, all these methods are tied to very specific mining algorithms because the information they share to construct global models are specific to those algorithms. For example, the method in Kargupta and Park (2004) shares decision trees represented using Fourier coefficients, and only works for decision trees. Thus numerous distributed algorithms are developed, each suited for just one of the corresponding mining algorithms thus not delivering an unified framework sought for, in real life business practices (Du *et al*., 2004).

A condensation approach has been proposed in Aggarwal and Yu (2004) which is targeted to be general. It tries to preserve data correlations, which is used by many mining algorithms like decision trees. However, data reduction or multiparty computations are not considered. Further, unlike the work presented here, the condensation approach is more concerned with hiding the identities of entities. Disclosure protection of the original data values is not good enough because the regenerated data values are very close to the original ones.

3 Proposed approach

This section first describes the fuzzy programming approach to select a minimal set of high-energy coefficients. It then discusses the solution in three cases of centralised data, horizontally partitioned data, and vertically partitioned data. Finally it proposes a random permutation protocol to enhance privacy.

3.1 Fuzzy programming approach for coefficient selection

The proposed method starts by taking DCT of the data row-wise. The core of the proposed method is to select high-energy coefficients across rows of transformed data. Let X_i denote a binary variable taking value 1 when a transform coefficient i is pruned and 0 otherwise. Also let n denote the number of rows, m denote the number of attributes, and W_{ij} denote the percentage energy stored by coefficient i in row j (it equals the energy of coefficient i in row j divided by the total energy of row j).

Problem 1 *The problem of pruning the maximal set of low energy coefficients with a certain maximum possible percentage energy loss of ζ for every row can be defined as the following Integer Linear Programming (ILP) problem:*

$$Max \quad \sum_{i=1}^{m} X_i$$

$$S.t \quad \sum_{i=1}^{m} W_{ij} X_i \le \zeta \quad 1 \le j \le n.$$

In Problem 1, users can specify a threshold (upper bound) ζ between 0 and 1 for the allowed energy loss. Problem 1 tries to maximise the number of pruned coefficients (*i.e.*, maximise privacy and data reduction through coefficient suppression), and ensures that the percentage energy loss for any of the rows does not exceed ζ. However, specifying an appropriate value for ζ is difficult. If the value is too small, the solution might have all $X_i = 0$, *i.e.*, none of the coefficients are pruned, and thus there is no privacy and data reduction. Furthermore, with one constraint for each row, large datasets will give enormous number of constraints, making the solution increasingly infeasible because ILP is NP complete and difficult to solve in general.

To solve the problem of setting ζ, users can specify a range $[b, b+p]$ instead of a fixed value for ζ and their choice function. Essentially ζ is replaced with a fuzzy variable $\tilde{\zeta}$. Users will be 100% satisfied if the actual maximal energy loss is no more than b, and will be 0% satisfied if the actual maximal energy loss is more than $b+p$. If the actual maximal loss falls in between b and $b+p$, users will be satisfied to a certain degree. For example, when $b = 0.1$ and $p = 0.2$, and the actual maximal energy loss across all rows is 0.2, users will be satisfied to a certain degree because the actual maximal energy loss falls in [0.1, 0.3].

The degree of satisfaction can be quantified using a membership function over the fuzzy variable $\tilde{\zeta}$. Following the most common practice in fuzzy programming literature, this paper uses linear membership functions to keep the formulations simple.

To eliminate the problem of too many constraints, this paper approximates all $W_{ij}, 1 \le j \le n$ of the i-th coefficient with another fuzzy variable $\tilde{W}_i$. The upper and lower bounds for W_i can be computed as the upper and lower bounds of $W_{ij}, 1 \le j \le n$. Let $[a_i, a_i+d_i]$ denote the range of $\tilde{W}_i$. This leads to the fuzzy programming problem as follows.

Problem 2 *The problem of retaining the minimal set of high-energy coefficients using fuzzy programming is given by:*

$$Max \quad \sum_{i=1}^{m} X_i$$

$$S.t \quad \sum_{i=1}^{m} \tilde{W}_i X_i \le \tilde{\zeta}.$$

This is very similar to the fuzzy programming problem depicted in Gasimov and Yenilmez (2002) excepting the fact that the X_i variables here are integers and thus the membership functions involving X_i will exist at discrete points and will not be continuous. However this will not change the problem in this context as is intuitively apparent.

In order to solve the problem, two membership functions need to be defined, one for the objective function in Problem 2, and the other for the constraint. The first membership function quantifies the degree of satisfaction of users over data reduction, that is, the higher the sum of X_i (*i.e.*, the more coefficients are pruned), the better the privacy and data reduction. The second membership function quantifies the degree of user satisfaction over energy retention, that is, the lower the energy loss (the left hand side of the constraint in Problem 2), the higher the degree of satisfaction.

To define the first membership function for the objective function, the upper and lower limits Z_u and Z_l of the objective function (sum of X_i) are evaluated using the following two integer linear programming problems:

$$Z_u = Max \sum_{i=1}^{m} X_i$$

$$S.t \quad \sum_{i=1}^{m} a_i X_i \le b + p$$

and

$$Z_l = Max \sum_{i=1}^{m} X_i$$

$$S.t \quad \sum_{i=1}^{m} (a_i + d_i) X_i \le b.$$

The first ILP problem finds the upper bound of the objective function because the constraint is relaxed by using the lower bounds (a_i) of energy loss of each coefficient and the upper bound ($b+p$) of the energy loss threshold. Similarly, the second ILP problem finds the lower bound of the objective function by strengthening the constraint using the upper bound (a_i+d_i) of energy loss and lower bound (b) of energy loss threshold. These two ILP problems can be solved efficiently by sorting a_i (or a_i+d_i) in ascending order, respectively, and selecting the first few that adds up to less than or equal $b+p$ (or b).

Thus the membership function μ_G for the objective function is defined as following:

$$\mu_G = \begin{cases} 0 & if \quad \sum_{i=1}^{m} X_i < Z_l \\ \dfrac{\sum_{i=1}^{m} X_i - Z_l}{Z_u - Z_l} & if \quad Z_l \le \sum_{i=1}^{m} X_i < Z_u \\ 1 & if \quad \sum_{i=1}^{m} X_i \ge Z_u \end{cases}.$$

That is, users will be 100% satisfied if the objective function is greater than Z_u, that is, the number of pruned coefficients reaches the upper bound. They will be 0% satisfied if the objective function is less than Z_l, that is, the number of pruned coefficients is below

the lower bound. Finally they will be satisfied to a certain degree when the number of pruned coefficients is between the upper and lower bound, and this degree follows a linear function as:

$$\frac{\sum_{i=1}^{m} X_i - Z_l}{Z_u - Z_l}.$$

To define the membership function for the constraint, this paper first computes an upper bound and a lower bound of the energy loss (left hand side of the constraint). Clearly, the upper bound is:

$$\sum_{i=1}^{m} (a_i + d_i) X_i.$$

This is because a_i+d_i is the upper bound of energy loss for pruning coefficient i. Similarly, the lower bound of energy loss is:

$$\sum_{i=1}^{m} a_i X_i.$$

Thus the actual energy loss always exceeds b+p if:

$$b + p < \sum_{i=1}^{m} a_i X_i.$$

Thus users will not be satisfied in this case because they have specified that the maximal energy loss shall be in the range of [b, b+p]. Similarly, users will be 100% satisfied if:

$$b \geq \sum_{i=1}^{m} (a_i + d_i) X_i.$$

That is, the maximal possible energy loss is less than b. The satisfaction of users follows a linear function in between these two cases. Thus the membership function μ_c for the constraint in Problem 2 is as follows:

$$\mu_C = \begin{cases} 0 & \text{if} \quad b + p < \sum_{i=1}^{m} a_i X_i \\ \dfrac{b + p - \sum_{i=1}^{m} a_i X_i}{\sum_{i=1}^{m} d_i X_i + p} & \text{if} \quad \sum_{i=1}^{m} a_i X_i \leq b + p < \sum_{i=1}^{m} (a_i + d_i) X_i + p. \\ 1 & \text{if} \quad b \geq \sum_{i=1}^{m} (a_i + d_i) X_i \end{cases}$$

Following the principle in Bellman and Zadeh (1970) and Gasimov and Yenilmez (2002) that the intersection of two crisp sets is equivalent to the minimum of membership functions of two fuzzy sets, the following membership function μ_D represents the overall satisfaction of users:

$$\mu_D = Min(\mu_G, \mu_C).$$

For example, consider a selection of X_i such that μ_G equals 0.5 and μ_C equals 0.4, meaning users are 50% satisfied with the objective and 40% satisfied with the constraint. The overall user satisfaction is 40%.

Now the goal is to maximise the overall satisfaction μ_D:

$$\underset{\mathbf{X}i,1\le i\le m}{Max}\ (\mu_D) = \underset{\mathbf{X}i,1\le i\le m}{Max}\ Min(\mu_G, \mu_C).$$

Let the value of μ_D be Ω. Thus:

$$\Omega \le \mu_G \ and\ \Omega \le \mu_C \ \text{ because } \ \mu_D = Min(\mu_G, \mu_C).$$

Thus Problem 2 can be converted to the following crisp mixed integer non-linear programming problem.

Problem 3

$$Max \quad \Omega$$

$$S.t \quad \Omega \le \frac{\sum_{i=1}^{m} X_i - Z_l}{(Z_u - Z_l)}$$

$$\Omega \le \frac{b + p - \sum_{i=1}^{m} a_i X_i}{\sum_{i=1}^{m} d_i X_i + p}$$

$$\sum_{i=1}^{m} X_i \le m - 1$$

$$0 \le \Omega \le 1$$

$$X_i \in \{0,1\}.$$

The constraint $\sum_{i=1}^{m} X_i \le m-1$ makes sure that not all coefficients are pruned under any circumstances. Problem 3 is typically non-convex optimisation and is difficult to solve. However there exist efficient algorithms like the outer approximation (Fletcher and Leyffer, 1994) to tackle such problems and give excellent approximate solutions even for a moderately large number of variables and constraints.

3.2 Solution to centralised database case

Figure 3 gives the pseudo code for the centralised scenario. Let m be the number of attributes and n be the number of rows. The time complexity of DCT is $O(nm\log m)$ (Oppenheim and Schafer, 1999). The time to compute bounds a_i and d_i is O(mn). Problem 3 is a mixed integer non-linear programming problem with m variables and 5 constraints. Suppose Problem 3 takes time O($T(m)$), the overall time complexity is O(mn log(m) + $T(m)$).

Figure 3 Algorithm for processing centralised data

Algorithm 1
Input parameters: b and p
Process(b, p)
1. For each row j=1 to n
2. transformed_data(j, 1 to m)←DCT(row(j))
3. End
4. For each coefficient i=1 to m
5. a_i=Minimum over all rows(transformed_data(1 to n,i))
6. a_i+d_i= Maximum over all rows(transformed_data(1 to n,i))
7. End
8. Solve Problem 3
9. Choose coefficients given by X_i=0
10. Randomly shuffle the coefficient indexes
11. Send selected coefficients from all rows following this order to the third party
12. End Process

3.3 Solution to horizontally partitioned database case

The horizontally partitioned case is an extension to the centralised case. This is attributed to the fact that maximum and minimum of all the coefficients globally, are basically the maximum and minimum of the maximum and minimum of each partition's data. Thus each site computes their own bounds and sends them to the server. The server then computes a global bound based on received bounds, and solves Problem 3. The time complexity is the same as the centralised case. Suppose μ coefficients are selected, k partitions are involved, the communication cost is O($\mu n + mk$), where each partition sends 2m bounds, and μn coefficients are sent to the server.

3.4 Solution to vertically partitioned database case

The solution to vertically partitioned case is based on the linearity property of Fourier related transforms that is, given two sequences X and Y:

$$DCT / DFT(X + Y) = DCT / DFT(X) + DCT / DFT(Y).$$

A zero padding scheme is used to make the datasets of equal dimension for classification. Each partition will contain the attributes included in that partition, plus zeros in the attributes appearing in all other partitions.

After this zero padding, all partitions have the same number of attributes. Using the linearity property of DCT, the sum of the i-th coefficient from the transformed padded partitions equals to the i-th coefficient computed over transformed vertically concatenated data. However, the sum of local maximum and minimum values of each partition may not be global maxima or minima as in Algorithm 2. Thus for each partition, two constraints would replace the first two constraints in Problem 3 using the bounds computed for that partition. The problem has now $2k+3$ constraints (two constraints per partition plus last three fixed constraints). Let the time to solve this non-linear mixed integer programming be $T'(m,k)$. Zero padding can be done in O(nm) time. Thus the time complexity is $O(mn + T'(m,k))$, and communication cost is O($\mu n + mk$).

3.5 The random permutation protocol to boost privacy

Since some coefficients are pruned, the third party cannot reconstruct the original data values exactly. Since DCT is a linear transform, getting back the original data is equivalent to solving a set of equations with more variables than equations which is not feasible. In the centralised database case, privacy can be further enhanced to a great degree simply by randomly permuting the selected coefficient indexes and not revealing the number of attributes to the third party. Without knowing the number of attributes and the order of coefficients, it is extremely difficult for the third party to reconstruct the original data. Let μ be the number of coefficients sent. The probability of guessing the right combination is given by:

$$p = \frac{1}{\sum_{i=\mu}^{m'} {}^{i}P_{\mu}}.$$

${}^{i}P_{\mu}$ is the number of permutations of i items taking μ items at a time and m' is the maximum possible number of attributes the third party guess. This probability is almost zero even for moderate m'.

The third party may try to prune some combinations when the bounds of some attributes are known to him. However, this approach is problematic. For example, suppose the third party knows the DCT coefficients for sequence 5000, 10 000, 50 000 but does not know the correspondence between coefficients and their indexes. Now assume that the third party reconstructs data using coefficients in the order 1, 3, and 2. The reconstructed data is 18 780, 47 647, and −1427. Now consider a case where the third part knows that the numbers are salary fields of three employees. The third party also knows that salary cannot be negative, thus, he can reject this permutation. However, if the third coefficient (the smallest) is pruned, the reconstructed data using the correct permutation of the first two coefficients will be −833, 21 667, and 44 167, which also contains a negative number.

Further, the above process is extremely expensive as the third party cannot eliminate a combination till it actually inverts it. If there are multiple permutations that satisfy the known bounds, it will also be extremely difficult for the third party to figure out the correct permutation.

In the horizontally and vertically partitioned cases, if the third party is the sole coordinator, then all sources must send coefficient with the same indexes as requested by the server. Thus the third party has to know the correspondence between indexes and coefficients and can potentially use this information to approximately reconstruct the original data. This situation can be made more secure very much like the one in centralised case by introducing a minor cryptographic tweak as following. The data sharing parties agree on a known, random permutation order which is not known to the server and send coefficients permuted in that order to the server. However, in the distributed cases, the number of attributes will be known to the third party as opposed to the centralised case. This is because the bounds of coefficients are communicated to the server for framing the optimisation problem. This slightly increases the chance of breaking the permutation protocol described above when compared to the centralised case. However, even a moderate number of attributes will give enough permutations

to make the discovery process prohibitively expensive as can be justifiably observed. The experimental section will present results that demonstrate that the approach preserves privacy to a high degree in the centralised case and the distributed cases with the permutation protocol. Privacy is also maintained to an appreciable degree in the worst case when the third party finds out the permutation, due to the pruning of coefficients and thereby reducing the ratio of number of equations to variables in the set for reconstructing original data.

4 Experimental evaluation

This section presents experimental evaluation of the proposed methods against existing methods. Section 4.1 describes the setup. Section 4.2 describes how to select parameters b and p for the proposed approach. The results for centralised case are presented in Section 4.3. The results for horizontally partitioned case and for vertically partitioned case are presented in Section 4.4 and Section 4.5, respectively. Section 4.6 reports the worst case privacy.

4.1 Setup

The experiments were conducted on a machine with Pentium 4, 3.4 GHz CPU, 4.0 GB of RAM, and running Windows XP. All algorithms were implemented using Matlab 7.0. General Algebraic Modelling System (GAMS) release 2.5 (GAMS Development Corporation, 1998) with DICOPT solver (Grossmann *et al*., 2003) was used for solving the mixed integer non-linear programming formulations.

4.1.1 Datasets

The experiments were run over two real datasets and one synthetic dataset. The two real datasets were Iris and Pendigits, both obtained from UCI Machine Learning Repository (Hettich *et al*., 1998). The synthetic data contained ten clusters, each generated using a multi-dimensional Normal distribution. Table 1 reports the properties of these data sets. For classification, 20% of data was randomly selected as the testing data, and the rest was used as training data.

Table 1 Properties of datasets

Properties	*Iris*	*Pendigits*	*Synthetic*
Number of attributes	4	16	50
Number of records	150	7494	10 000
Number of classes	3	10	10

4.1.2 Data mining algorithms

K-means clustering and k-nearest neighbour classification were used in experiments. k was set to 5 in KNN.

4.1.3 Privacy preserving algorithms

A DCT-F algorithm was implemented using DCT based on algorithms proposed in Section 3. The following four algorithms were also implemented and compared against DCT-F:

1 *DCT-R* – it is the same as DCT-F except that coefficients were selected randomly.

2 *Rand-N* – this algorithm adds to original data a random noise following Gaussian distribution with mean = 0. The standard deviation was varied in the experiments to generate different degree of privacy.

3 *Rand-U* – this algorithm adds to original data a random noise following uniform distribution with mean equals 0. The interval of the uniform distribution was varied to generate different degree of privacy.

4 *Rand-P* – this is the random projection method proposed in Liu *et al*. (2006) and Oliveira and Zaïane (2004).

4.1.4 Setup for horizontally partitioned case

The number of partitions was varied from 2 to 5. Data records with different class labels were uniform randomly distributed among these sites.

4.1.5 Setup for vertically partitioned case

The number of partitions was varied from 2 to 3 and data columns were randomly distributed to each partition.

4.1.6 Privacy measure

Three approaches have been proposed in the literature to measure privacy: the first using confidence interval (Agrawal and Srikant, 2000), the second using information theory (Agrawal and Aggarwal, 2001), and the third based on the notion of privacy breach (Evfimievski *et al*., 2003). However, the information theory approach is inappropriate for K-means and KNN classification because Euclidean distance is based on individual data values, while information theory only considers the distribution of values. Privacy breach based methods consider the worst cases, but here interest lies in the average case. Thus this paper uses the confidence interval method proposed in Agrawal and Srikant (2000) to measure privacy. If a transformed attribute x can be estimated with $c\%$ confidence in the interval $[x_1, x_2]$, then the privacy equals:

$$\frac{x_2 - x_1}{\max x - \min x}.$$

Ninety-five percent confidence interval was used in experiments.

For DCT-F and DCT-R, this paper considers two cases: the average case when the third party does not figure out the correct number of attributes and the permutation of coefficients, and the worst case when the third party does figure out the number of attributes and the permutation of coefficients. In the average case, the third party randomly guesses the number of attributes and a permutation of coefficients, and

reconstructs the data using this permutation. The privacy is computed by comparing the reconstructed data and the original data. This process is repeated for 20 times and the average of privacy is reported. In the worst case, the privacy is computed by comparing the original data and the reconstructed data with correct number of attributes and permutation of coefficients. The results section reports the average case privacy in Section 4.2, 4.3, 4.4, and 4.5 and reports the worst case privacy in Section 4.6.

For Rand-P, it is also difficult for the third party to figure out the projection matrix because it is generated randomly. Thus privacy is computed by directly comparing the transformed data with the original data, assuming the missing columns contain zeros.

4.1.7 Data mining quality measure

In this paper, the quality of classification is measured by accuracy. The quality of clustering is measured using the *F* measure that is widely used in information retrieval (Rijsbergen, 1979).

4.2 Guideline for selecting b *and* p

For DCT-F, users need to select the range [*b*, *b*+*p*] for the energy loss threshold (*i.e.*, the upper bound of energy loss). This section illustrates a guideline to select *b* and *p*.

4.2.1 Fixing p*, varying* b

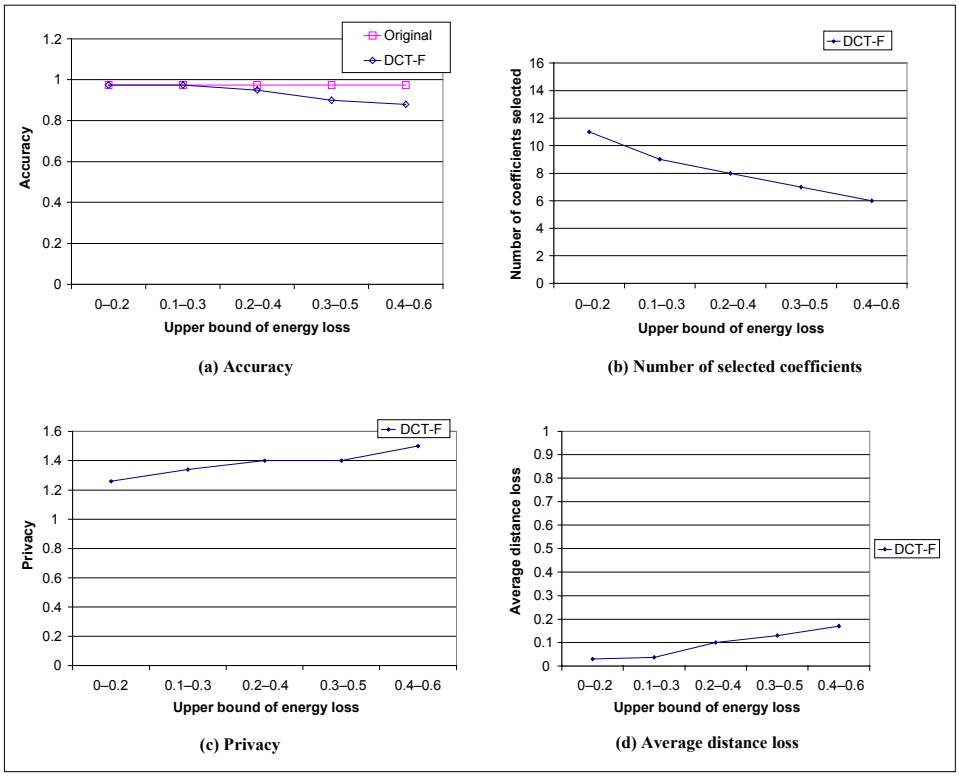
The first set of experiments set the value of *p* to 0.2, and varied the value of *b*. Figure 4 (a), (b), (c), and (d) report the accuracy of K-Nearest Neighbour (KNN) over data transformed by DCT-F, the number of coefficients being selected (not the number of pruned coefficients), the degree of privacy for DCT-F with the permutation protocol, and the average loss of distance over Pendigits data, respectively. The accuracy of KNN over the original data is also plotted as a baseline for comparison in Figure 4(a). The results for Iris and Synthetic data set are similar and omitted. The results for K-means clustering are also similarly appreciable and omitted. The distance loss between record *i* and *j* is computed as follows:

$$| d_{ij} - d'_{ij} | / d_{ij} \quad where \quad i \neq j.$$

Here d_{ij} is the Euclidean distance between record *i* and record *j* in the original data, and d'_{ij} is the corresponding distance in the transformed data. The loss is normalised by d_{ij} and then averaged over all pairs or records. In case d_{ij} equals zero, the loss is normalised by the average of d_{ij}.

As $b = 0$ and $p = 0.2$, the upper bound of energy loss was between 0 and 0.2, and users were 100% satisfied when the maximal energy loss was zero and 0% satisfied when the maximal energy loss reached 0.2. The results showed that under this setting, 11 out of 16 coefficients were selected, and KNN achieved the same accuracy on the transformed data as on the original data. As *b* increased, the maximal allowed energy loss also increased, and fewer coefficients were selected. The accuracy of KNN started to drop slightly. This drop can be best explained by Figure 4(d), which showed that the average loss of distance was very low (about 3%) as $b = 0$, and increased as *b* increased. The average distance loss was very low for small *b* values, and was below 20% for even large *b* values, showing that DCT-F preserves Euclidean distance to a large degree.

Figure 4 Classification for Pendigits data, varying b with $p = 0.2$



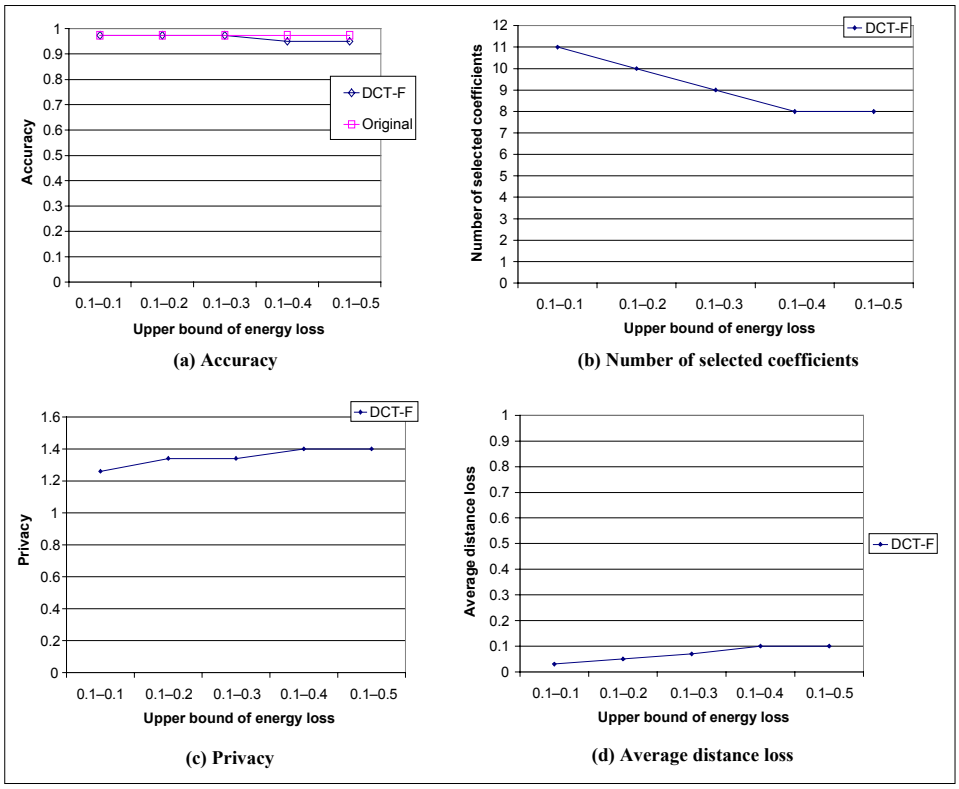
The results also showed that the degree of privacy using DCT-F with the permutation protocol was very high, due to pruning of coefficients as well as the fact that the third party did not know the number of attributes and the correspondence between coefficients and their indexes.

4.2.2 *Fixing* b*, varying* p

The next set of experiments fixed the value of b to 0.1, and varied the value of p. Figure 5 (a), (b), (c) and (d) report the accuracy of KNN over data transformed by DCT-F, the number of coefficients being selected, the degree of privacy for DCT-F with the permutation protocol, and the average loss of distance over Pendigits data, respectively. The results for Iris and Synthetic data as well as clustering them with k-means are similar and omitted.

The results were very similar to the case of fixing p but varying b. As $b = 0.1$ and $p = 0$, 11 out of 16 coefficients were selected, and KNN achieved the same accuracy on the transformed data as on the original data. As p increased, the maximal allowed energy loss also increased, and fewer coefficients were selected. The accuracy of KNN started to drop slightly. Figure 5(d) showed that the average loss of distance was very low (about 3%) as $p = 0$, and increased with p. The results also showed that the degree of privacy using DCT-F with the permutation protocol was pretty high.

Figure 5 Classification for Pendigits data, varying p with $b = 0.1$



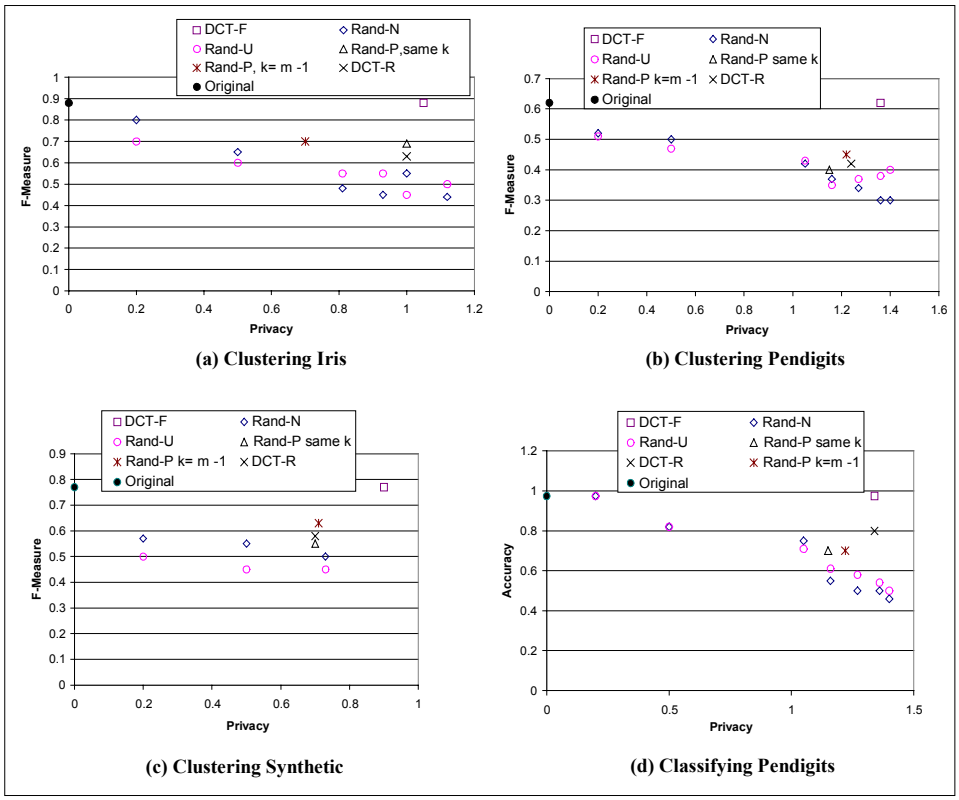
Overall, the results showed that the quality of mining on the transformed data was very close to the quality of mining on the original data for small b and p values. The degree of privacy in the average case was high for all b and p values. Thus, users can select relatively small b and p values. In the data sets used in this paper, $b = 0.1$ and $p = 0.2$ is a good selection because the mining quality on the transformed data is almost the same as the mining quality on the original data, the degree of privacy is high (over 100%), and the data size is reduced to 56% of the original data. Thus in the following experiments, b is set to 0.1 and p is set to 0.2.

4.3 Results for centralised database case

This section compares the results of DCT-F with the other four algorithms for the centralised database case.

This section compares DCT-F with the four existing algorithms. Figure 6 reports the mining quality and privacy of these algorithms. The original data is also shown as a baseline. The standard deviation of noises added by Rand-U and Rand-N were varied to generate various degree of privacy. The same number of coefficients was used for DCT-R and DCT-F. Rand-P needs to specify k, the number of columns after projection. Two k values were used: k equals the number of coefficients selected by DCT-F (thus both DCT-F and Rand-P reduced the data to the same size), and k equals total number of columns (m) minus 1, which is the largest k value that can be used to reduce data size.

Figure 6 Privacy versus mining quality



The results showed that DCT-F always led to better mining quality than Rand-N and Rand-U when generating data with similar degree of privacy. The mining quality of DCT-F was almost the same as the mining quality over the original data. The random noise added by Rand-N and Rand-U distorted the Euclidean distances, leading to poor mining quality. In all experiments, Rand-N and Rand-U only led to good mining quality when the degree of privacy was very low (around 20%). Further, unlike DCT-F, random perturbation methods also do not reduce the data size.

The results also showed that DCT-F led to better mining quality than Rand-P for both k values, while providing similar degree of privacy. The results also showed that DCT-F led to better mining quality than DCT-R, which randomly selected coefficients. This shows that the fuzzy programming approach, to find the right coefficients to prune, is effective.

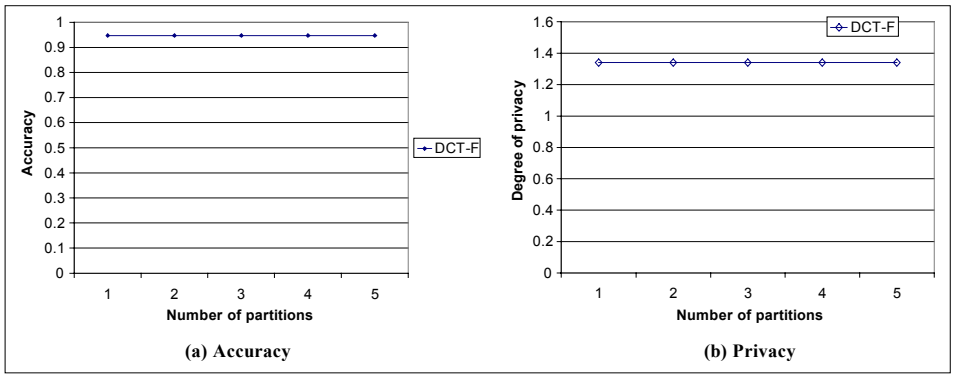
Degree of data reduction

The transformed data using DCT-F with b varying from 0.1 to 0.2 was 56%, 50% and 36% of the original data size for Pendigits, Iris, and Synthetic data respectively, yet giving almost the same quality of mining as the original data.

4.4 Results for horizontally partitioned case

This section describes the results of KNN classification for the horizontally partitioned case. The results for clustering are similarly quite appreciable and not reported. Figure 7(a) reports the accuracy of KNN classification using the distributed version of DCT-F for the Pendigits data when $b = 0.1$ and $p = 0.2$. Figure 7(b) reports the degree of privacy.

Figure 7 Horizontally partitioned case, classifying Pendigits

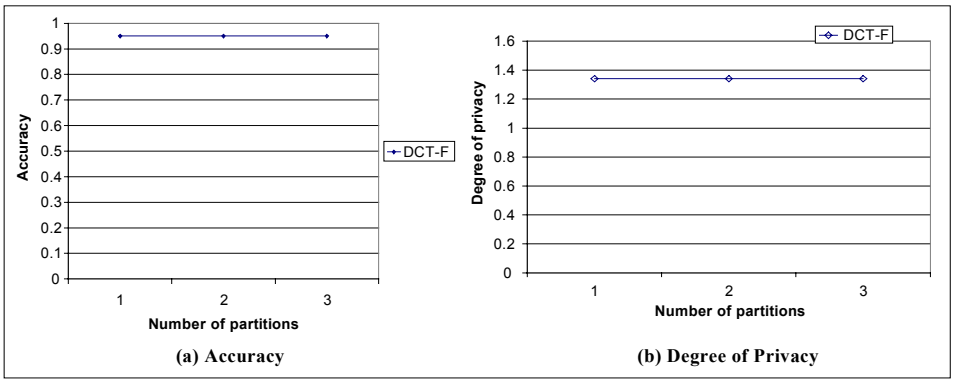


The results showed that the accuracy and degree of privacy were the same for different number of partitions. This is expected because the energy bounds of horizontally partitioned data are the same as centralised data.

4.5 Results for vertically partitioned case

This section describes the results for KNN classification for the vertically partitioned case. The results for clustering are also similar and not reported. Figure 8(a) reports the accuracy of KNN classification using distributed version of DCT-F for the Pendigits data when $b = 0.1$ and $p = 0.2$. Figure 8(b) reports the degree of privacy with the permutation protocol. The results showed that the accuracy and degree of privacy were the same for different number of partitions. Note that in the centralised case, the accuracy over the transformed data was almost the same as the accuracy over the original data. Thus DCT-F also works well for vertically partitioned case.

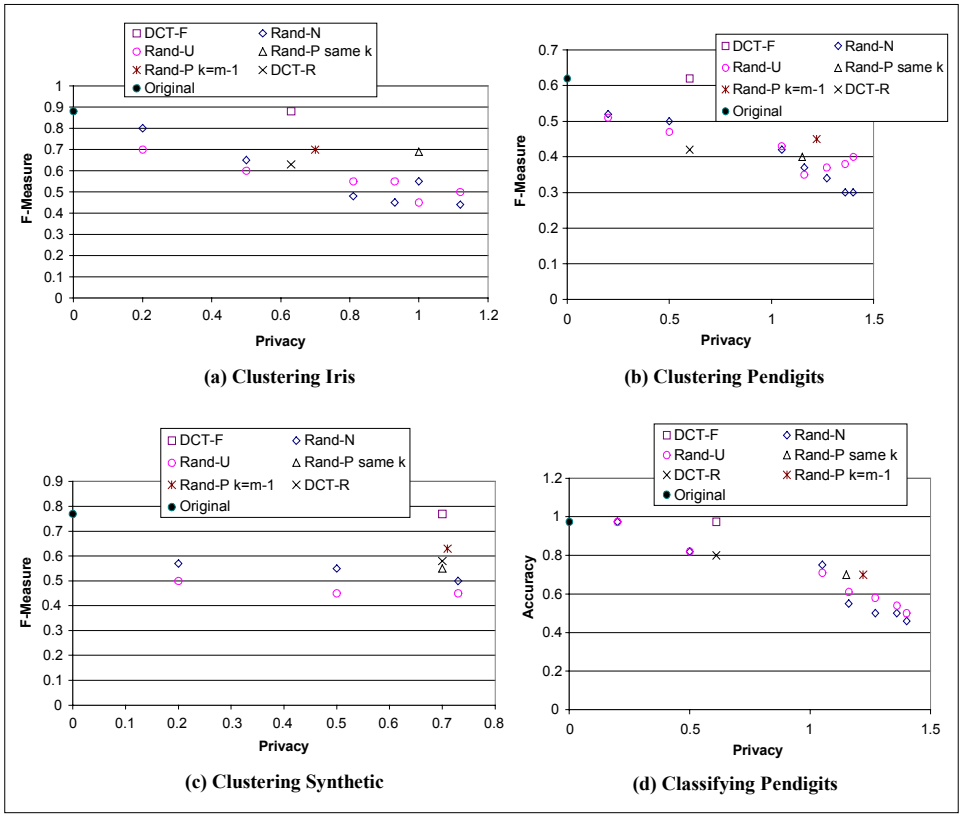
Figure 8 Results for vertically partitioned case, classifying Pendigits



4.6 Worst case privacy

The proposed DCT-F method permutes the selected coefficients such that it is difficult, if not impossible for the third party to reconstruct the data. The previous sections have reported the degree of privacy when the third party does not discover the permutation. This section reports the worst case privacy when the third party discovers the correct permutation and number of coefficients and reconstructs the data assuming pruned coefficients to be 0. Figure 9(a), (b), (c) and (d) report the mining quality and the degree of privacy for various algorithms over various data sets.

Figure 9 Worst case privacy and quality of mining



The results showed that the worst case privacy using DCT-F was lower than the average case privacy, because unlike when the permutation is not known, the third party could use inverse DCT to reconstruct the data. However, DCT-F still achieved considerable degree of privacy (50%–70%) because of the pruning of coefficients. The results showed that DCT-F still achieved significantly better accuracy than Rand-U and Rand-N for similar degree of privacy. Note that the accuracy of DCT-F was very close to the benchmark case (*i.e.*, over original data). Further, DCT-F also reduced the data size, while Rand-U and Rand-N did not. The results also showed that DCT-F achieved higher accuracy than Rand-P, but often with a lower degree of privacy.

In the experiments, DCT-F took less than one second in all data sets tested.

5 Conclusion

This paper proposes an integrated approach for data reduction and privacy for distance-based mining algorithms using Fourier-related transforms. It develops a novel fuzzy programming formulation to achieve the optimal tradeoff between data reduction and preserving of Euclidean distances while incorporating data privacy. Experimental results demonstrate that the proposed approach leads to much better mining quality than the existing random perturbation and random projection approaches given the same degree of privacy in both centralised and distributed cases. The novelty of the approach is an attempt to bring a number of privacy preserving mining techniques and scenarios sharing the common theme under the same umbrella. In the future the plan is to investigate how to provide probabilistic privacy and accuracy guarantees using the approach, and to incorporate other algorithms, which can be related indirectly to Euclidean distance similarity measure, into the framework.

References

Aggarwal, C.C. and Yu, P.S. (2004) 'A condensation approach to privacy preserving data mining', *EDBT 2004, 9th International Conference on Extending Database Technology*, Heraklion, Crete, Greece.

Agrawal, D. and Aggarwal, C.C. (2001) 'On the design and quantification of privacy preserving data mining algorithms', *PODS*, Santa Barbara, California.

Agrawal, R., Faloutsos, C. and Swami, A.N. (1993) 'Efficient similarity search in sequence databases', *4th International Conference of Foundations of Data Organization and Algorithms*.

Agrawal, R. and Srikant, R. (2000) 'Privacy preserving data mining', *SIGMOD*, Dallas, Texas, May.

Bellman, R.E. and Zadeh, L.A. (1970) 'Decision making in a fuzzy environment', *Management Science*, Vol. 17, December, pp.141–164.

Canz, T. (1999) 'Fuzzy linear programming for DSS in energy planning', *International Journal of Global Energy Issues*, Vol. 12, pp.138–151.

Cover, T.M. (1968) 'Rates of convergence for nearest neighbor procedures', *Inter. Conf. on Systems Sciences*.

Du, W. and Atallah, M.J. (2001) 'Secure multi-party computation problems and their applications: a review and open problems', *2001 Workshop on New Security Paradigms*, Cloudcroft, New Mexico.

Du, W., Clifton, C. and Atallah, M.J. (2004) 'Distributed data mining to protect information privacy', *NSF Information and Data Management (IDM) Workshop*.

Duda, R. and Hart, P.E. (1973) *Pattern Classification and Scene Analysis*, John Wiley & Sons.

Egecioglu, O.M., Ferhatosmanoglu, H. and Ogras, U. (2004) 'Dimensionality reduction and similarity computation by inner-product approximations', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, June, pp.714–726.

Evfimievski, A., Gehrke, J. and Srikant, R. (2003) 'Limiting privacy breaches in privacy preserving data mining', *PODS*.

Fletcher, R. and Leyffer, S. (1994) 'Solving mixed integer nonlinear programs by outer approximation', *Mathematical Programming*, Vol. 66, pp.327–349.

GAMS Development Corporation (1998) 'General algebraic modelling system', Washington, DC, http://www.gams.com/Default.htm.

Gasimov, R.N. and Yenilmez, K. (2002) 'Solving fuzzy linear programming problems with linear membership functions', *Turkish Journal of Mathematics*, Vol. 26, pp.375–396.

Grossmann, I.E., Viswanathan, J. and Vecchietti, A. (2003) 'DICOPT mixed integer non-linear programming solver', Engineering Research Design Center, Carnegie Mellon University, Pittsburgh, Pennsylvania, http://www.gams.com/dd/docs/solvers/dicopt.pdf.

Hettich, S., Blake, C.L. and Merz, C.J. (1998) *UCI Repository of Machine Learning Databases*.

Johnson, W.B. and Lindenstrauss, J. (1984) 'Extensions of Lipshitz mapping into Hilbert space', *Conference in Modern Analysis and Probability*.

Kahraman, C. and Ulukan, Z. (1999) 'Multi-criteria capital budgeting using FLIP', *3rd International Conference on Computational Intelligence and Multimedia Applications*.

Kargupta, H., Datta, S., Wang, Q. and Sivakumar, K. (2003) 'Random data perturbation techniques and privacy preserving data mining', *Knowledge and Information Systems*, Vol. 7, pp.387–414.

Kargupta, H. and Park, B.H. (2004) 'A fourier spectrum-based approach to represent decision trees for mining data streams in mobile environments', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 16, pp.216–229.

Liu, K., Kargupta, H. and Ryan, J. (2006) 'Random projection-based multiplicative data perturbation for privacy preserving distributed data mining', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 18, January, pp.92–106.

Oliveira, S. and Zaïane, O.R. (2003) 'Privacy preserving clustering by data transformation', *18th Brazilian Symposium on Databases*.

Oliveira, S. and Zaïane, O.R. (2004) 'Privacy-preserving clustering by object similarity-based representation and dimensionality reduction transformation', *Workshop on Privacy and Security Aspects of Data Mining (PSDM'04)*.

Oppenheim, A.V. and Schafer, R.W. (1999) *Discrete-Time Signal Processing*, Prentice-Hall.

Rijsbergen, C.J.V. (1979) *Information Retrieval*, Butterworths.

Zimmermann, H.J. (1978) 'Fuzzy programming and linear programming with several objective functions', *Fuzzy Sets and Systems*, Vol. 1, pp.45–55.