T-79.511 Special Course on Cryptology / Privacy Preserving Data Mining

Xavier RONDE-OUSTAU

December 14, 2003

Introduction

Data Mining aims, according to its own definition, to extract from large databases useful non-obvious knowledge and information. Thanks to technological breakthrough in matters of transmission, storage and processing of larger and larger amounts of data, the question of preserving privacy becomes a topical issue in many fields.
Data Mining is often perceived as a threat for privacy, but one should not forget all the advantages brought by data mining and data analysis. That is why it is important to develop techniques that would let the user get all benefits from data mining within keeping a high level of privacy for sensitive or confidential data.
The privacy requested by a person for an attribute can by of three orders:
- the given true value may not be revealed
- the given true value can be revealed
- the value should be modified (”noisy”)

One of the main applications of data mining is the use of decision-tree classifier. We will see later that only the probability distribution of the data are needed in order to construct such a classifier, since they are quite powerful regarding their rapidity and their accuracy. We will therefore focus on data preserving methods that provide a good approximation of the statistical distribution of the original data by the perturbed data.

In order to protect privacy of sensitive data, two methods are suggested in this paper: query restrictions and data perturbations. The query restriction can be tackled form different aspects, like limited the number of results, keeping record of previous queries and checking for each new query if there could be any compromise... The data perturbations methods can for example add noise to the values of the database, to the result of the query, or sampling the database or the result of the queries. The main problem with this method is that it doesn't provide reliable statistics as the original would do. The precision is given by the variance of the estimators provided to the user. When this variance is equal to zero, we have an exact disclosure: we know exactly a confidential attribute. When this variance is below a predetermined threshold, we have a partial disclosure. The aim of privacy preserving data mining is to limit as much as possible the number of these disclosures.

We will consider two methods for modifying the values of a field: - Value Class Membership. The set of values is partitioned into complementary sets called class. The returned value is then the class to which the true value belongs.
- Value Distortion. We add a noise to the true value and obtain the value returned to the user.

For the experiments presented in this paper, it will be assumed that unauthorized access to the system is impossible and some protection exist, like secure transmission channel, and secure storage.

In section 2, we will study privacy preserving methods, while section 3 will focus on reconstructing the original data construction from perturbed data. We will discuss in section 4 some techniques for building decision tree classifiers. An experimental evaluation of the accuracy of these methods will be presented in section 5, before a short conclusion in section 6.

1 Privacy-Preserving Methods

The privacy considered here will be that a user can give a modified value of sensitive attributes. The way data are perturbed is up to the user. We will consider the two methods stated above in the introduction.

Value Class Membership

The discretisation, which is also a kind of sampling, is a special case of value class membership. We partition the attribute into intervals which are not bound to be equal. For example, the age could be discretised this way:

$[age < 25, 25 \leq age < 35, 35 \leq age < 60, 60 \leq age]$

Value Distortion

This is when we add to the original value a random value whose distribution can be chosen freely. We will however concentrate on two different random distribution:

- uniform distribution: we choose a boundary a and the noise will be equally spread on this $[-a, a]$ interval.

- Gaussian distribution: this is a normal distribution with a mean equal to zero and a standard deviation of σ.

Quantifying privacy

We estimate with $c\%$ confidence that x belongs to the interval $[x_1, x_2]$. The privacy metric is then the interval width, $x_2 - x_1$. When the discretization step is equal to the uniform random distribution amplitude, we have obviously the same privacy. In order to increase privacy for the uniform value distortion method, we just need to increase the amplitude of the values that the random function can take. For discretisation methods, it is needed to increase the sample sizes, and therefore reduce their number. The accuracy of this model would be very damaged, therefore, the uniform random distribution is preferable. Gaussian random perturbation provides also better privacy results than the two other methods. The rest of the paper will therefore focus on this value distribution perturbation method.

2 Reconstructing the original distribution

The values in the database, $x_1, x_2, \ldots, x_n$ are considered as being realisation of the n independent identically distributed random variables $X_1, X_2, \ldots, X_n$, which all have the same distribution as random variable X. The same goes for the random perturbation distribution as random variable Y. The information available are the $x_i + y_i$ and the cumulative distribution, F_Y. The aim is to find F_X, cumulative function for X. This is what is called the *reconstruction problem.*

In order to adopt a lighter notation, $w_i = x_i + y_i$. The estimate of the posterior distribution F'_{X_1}, given that $X_1 + Y_1 = w = 1$ is by definition

$$F'_{X_1}(a) = \int_{-\infty}^{a} f_{X_1}(z|X_1 + Y_1 = w_1)dz$$

We assume we know the density functions f_X and f_Y for X and Y respectively. We use the Bayes' rule to expand this expression:

$$F'_{X_1}(a) = \int_{-\infty}^{a} \frac{f_{X_1+Y_1}(w_1|X_1 = z)f_{X_1}(z)}{f_{X_1+Y_1}(w_1)}dz$$

We try to have only f_{X_1} and f_{Y_1} .We develop the denominator:

$$F'_{X_1}(a) = \int_{-\infty}^{a} \frac{f_{X_1+Y_1}(w_1|X_1 = z)f_{X_1}(z)}{\int_{-\infty}^{\infty} f_{X_1+Y_1}(w_1|X_1 = z')f_{X_1}(z')dz'}dz$$

Y_1 is independent of X_1. Therefore $f_{X_1+Y_1}(w_1|X_1 = z) = f_{Y_1}(w_1 - x_1|X_1 = z) = f_{Y_1}(w_1 - z)$. The integral becomes:

$$F'_{X_1}(a) = \frac{\int_{-\infty}^{a} f_{Y_1}(w_1 - z)f_{X_1}(z)dz}{\int_{-\infty}^{\infty} f_{Y_1}(w_1 - z)f_{X_1}(z)dz}$$

Since $f_{X_1} \equiv f_X$ and $f_{Y_1} \equiv f_Y$

$$F'_{X_1}(a) = \frac{\int_{-\infty}^{a} f_Y(w_1 - z) f_X(z) dz}{\int_{-\infty}^{\infty} f_Y(w_1 - z) f_X(z) dz}$$

In order to obtain an estimate of F'_X, we average the sum of F'_{X_i}.

$$\begin{aligned} F'_X(a) &= \frac{1}{n} \sum_{i=1}^{n} F'_{X_i} \\ &= \frac{1}{n} \sum_{i=1}^{n} \frac{\int_{-\infty}^{a} f_Y(w_i - z) f_X(z) dz}{\int_{-\infty}^{\infty} f_Y(w_i - z) f_X(z) dz} \end{aligned}$$

We want to get the posteriori density function f'_X which is obtained by differentiating F'_X.

$$f'_X(a) = \frac{1}{n} \sum_{i=1}^{n} \frac{f_Y(w_i - z) f_X(z)}{\int_{-\infty}^{\infty} f_Y(w_i - z) f_X(z) dz} \qquad (1)$$

With sufficiently high number of samples (n being high enough), we can expect the posteriori function f'_X to be close to the original one, f_X.

The problem is that we don't know $f_X(a)$ since it is the quantity we want to estimate. The idea is then to use an iterative algorithm, with initial f_X^0 distribution being a uniform distribution. Then, we have the following recursive formula:

$$f_X^{j+1}(a) = \frac{1}{n} \sum_{i=1}^{n} \frac{f_Y(w_i - z) f_X^j(z)}{\int_{-\infty}^{\infty} f_Y(w_i - z) f_X^j(z) dz}$$

The stopping criterion needs to be defined. Since we don't know the distribution we have to reach, we can stop the algorithm when the difference between two consecutive f_X^j is below a predefined threshold, thinking that the real distribution is not fare from this one. Unfortunately, some empirical results show that this is not always true.

In order to speed the computation of the reconstructed distribution, we can use a partitioning method. That way, the distance between w_i and z is approximated with the distance between mid points of the intervals containing w_i and z. Secondly, $f_X(a)$ is approximated with the average of the density function over the interval in which a lies. The result is that the complexity of the computation is reduced from $O(n^3)$ to $O(n^2)$.

3 Decision Tree Classifiers over Randomized Data

3.1 Background

The data is partitioned into different classes. A decision tree classifier will recursively sort the data until each leaf of the tree is composed of data from the same class. Each node of the tree is a split point, splitting data into to sub trees according to a test.

There are usually two phases to build the decision tree classifier: a growth phase and a prune phase. In the growth phase, the tree is built by recursively partitioning the data until each partition contains members belonging to the same class. Once the tree has been fully grown, it is pruned in the second phase to generalize the tree by removing dependence on statistical noise or variation that may be particular only to the training data. With this, we want to avoid having exception that would mask a global model valid in general, even if in some particular cases it is not valid.

The choice of where the splitting point should be placed is done by using the /emphgini index, which is defined as follows (with S being the data to be partitioned):

$$gini(S) = 1 - \sum p_j^2$$

where p_j is the relative frequency of class j in S. The split of S into S_1 and S_2 is optimum when it minimises the $gini_{split}$ which is defined by the following formula:

$$gini_{split}(S) = \frac{n_1}{n} gini(S_1) + \frac{n_2}{n} gini(S_2)$$

where n_i is the number of records in child i and n the number of records at the considered node. In a general way, the gini method will always favourites to have the purest classes possible.

3.2 training using randomized data

Determining a split point The data is reconstructed before we need to find out an optimal split point. We only need statistics to determine this split point using the gini index method, and we have them. The candidate split points are either the boundaries of the intervals of the speed computation algorithm, or any midpoint between two attributes for the standard reconstruction method.

Partitioning the data If $I_1, \ldots, I_m$ are the intervals issued from the reconstruction process, if a split occurs at the boundary between I_p and I_{p+1}, the points from $I_1 \ldots I_p$ go to S_1 while the others go to S_2.

Reconstructing the original distribution Reconstruction can take place at different places.

- **Global** The distribution of each attribute is computed in the beginning, before classification. This algorithm is not computationally demanding.

- **ByClass** For each attribute, the data is first divided into classes, and the distributions are computed for each class. This is more computationally demanding than the Global scheme.

- **Local** This is done like the ByClass method, but instead of doing this only in the beginning, this is done at every nodes. This one takes of courses a lot more resources than the two other schemes.

4 Experimental Results

We are comparing the classification accuracy of the 3 different reconstruction methods. We take the original and the purely randomized data as benchmark in order to compare the improvements or gain brought by the reconstruction methods. I will not give details about these training data since for the purpose of this paper, the most important thing is to focus on the results given by the authors of the original paper.

The figure 5 of Agrawal and Srikant paper shows the accuracy of the algorithms for uniform and Gaussian perturbation for privacy level of 25% and 100% for the 5 functions defined in their test. Local and ByClass algorithms perform at 5% accuracy for functions 1, 4 and 5, and at 15% for the two other functions. In general, the Global algorithm performs worse than the others.

The figure 6 tends to analyse the evolution of accuracy of the ByClass algorithm when requested privacy increases. There is quite a big difference between uniform and Gaussian randomisation when no reconstruction is applied, the uniform one performing worse than the Gaussian one. But when we use corrected data obtained from the reconstruction technique mentioned above, both are quite close to each other and close to the original, meaning that we get quite a good accuracy for the classifier.

Conclusion

We have seen that using randomization, we can reconstruct the with a good accuracy the original distribution. The sensitive values remain non-revealed, preserving privacy. From the experiences, it has been found that ByClass and Local reconstruction are effective methods since at even 100% privacy we have accuracy between 5% and 15% of the original accuracy. Local performs slightly better than ByClass but at a cost of a far higher complexity. And finally, with reconstruction, Gaussian and uniform perturbations perform similarly. Gaussian provides however higher privacy at higher confidence threshold than uniform, but it might be difficult to explain to users what this randomization does, uniform randomization being more easily understandable.

Reference

R. Agrawal and R. Srikant. Privacy preserving data mining. In ACM SIGMOD Conference on Management of Data, pages 439450, Dallas, Texas, May 2000.