NRL Release Number 04-1221.1-0054

Statistical Sensitive Data Protection And Inference Prevention with Decision Tree Methods

LiWu Chang *

Abstract

We present a new approach for protecting sensitive data in a relational table (columns: attributes; rows: records). If sensitive data can be inferred by unauthorized users with non-sensitive data, we have the inference problem. We consider inference as correct classification and approach it with decision tree methods. As in our previous work, sensitive data are viewed as classes of those test data and non-sensitive data are the rest attribute values. In general, however, sensitive data may not be associated with one attribute (i.e., the class), but are distributed among many attributes. We present a generalized decision tree method for distributed sensitive data. This method takes in turn each attribute as the class and analyze the corresponding classification error. Attribute values that maximize an integrated error measure are selected for modification. Our analysis shows that modified attribute values can be restored and hence, sensitive data are not securely protected. This result implies that modified values must themselves be subjected to protection. We present methods for this ramified protection problem and also discuss other statistical attacks.

1 Introduction

Information sharing and data disclosure have led to unprecedent demand for effective sensitive data protection methods. If sensitive data can be inferred by unauthorized users from non-sensitive data, we have the inference problem. In this paper, we apply the decision tree method ([5]) to inference prevention and sensitive data protection. The decision tree method conveniently provides a more localized description of data records. We assume that the data set is in the form of a relational table (columns: attributes; rows: records) and contains two parts: one is the training and the other is the test (Table 1). The decision tree method was first discussed in [1], where sensitive data were represented as values of the class_label (i.e., the class attribute) of the test data. However, sensitive data may not be restricted to one particular attribute and the class_label in many data sets is not specified. We consider the case where sensitive data are distributed over the entire data set. We extend the decision tree method to handle distributed sensitive data.

*Correspondence: Mail code 5540, Naval Research Laboratory, Washington, DC 20375 USA. lchang@itd.nrl.navy.mil

2 Inference Problem

We consider a simple two-leveled security protocol which has *High* and *Low* users. The High users (e.g., the database manager) view the entire database, and the Low users share the High view with the exception of any confidential data. When data are shared, High releases some of the non-sensitive data to Low. In the pre-processing of data release, sensitive data are replaced by "?"s. It is well-known that to prevent inference, removal of sensitive data alone is insufficient and modification of some non-sensitive data (e.g., blocking) is necessary (e.g., [3]).

Inference prevention proceeds as follows ([2]). High generates rules from the available data set, and then determines whether there is inference based on those rules. If the inference is excessive, then it implements a protection plan to lessen the inference (i.e., decides to modify by deleting certain data from the database as it appears to Low). The output of our inference model is the data set that can be released to Low. Our goal is to make modifications as parsimoniously as possible and thus avoid imposing unnecessary changes which lessen functionality.

Table 1: Relational Table for Evaluation. A_j denotes the jth attribute and the "?" denotes an unknown value, a piece of confidential datum, or a previously modified value.

key	A1	A2	..	A_k	..	A_M	class_label
training data	..	?	..	..	..	..	..
	?	..	..	?	..	?	..
	..	..	..	..	..	..	..
testing	..	..	..	?	..	..	?
	..	?	..	..	..	..	?

3 Decision Tree Method

The class_label attribute in conventional decision tree methods is deterministic. To deal with inference in the presence of distributed sensitive data, any attribute may be considered as the class_lable (thereby the original class_label becomes an ordinary attribute.)

As in our previous work ([1]), sensitive data are viewed as classes of those test data and non-sensitive data are the rest attribute values. We consider inference as correct classification - the lower the correct classification, the higher the security of the data will be. To prevent inference, we increase (decrease) the classification error (correct classification) by modifying the set of attribute values of non-sensitive data that yields the largest increase (decrease) in classification error (correct classification). We formalize the requirements in the next section.

4 Metric

Data modification is most likely to incur degradation of data performance. Important metrics in data modification are the effectiveness measure of sensitive data protection (E) and the measure of the loss of functionality (F) in a data set. In terms of the decision tree method, the effectiveness measure (w.r.t. to the current class_label) is determined by the classification error of the test data (i.e. the confidential data), while the measure of loss of functionality is a function of the classification error of the training data (i.e. the to-be-released data).

Suppose the jth attribute is posted as the class_label. Let the measure of protection effectiveness with respect to the jth attribute be denoted as E_j and the measure of the loss of functionality be denoted as F_j. The overall measure of E and F for the entire database are the function (e.g.,weighted average) of E_is and F_is. The measure of the loss of functionality F is usually has an upper bound of a given threshold υ (i.e., $F \leq \upsilon$) that represents the maximum level of information loss that users are willing to tolerate. With the definitions of E and F in mind, our optimization goal is to

$$\text{Minimize } E, \text{ while keeping } F \leq \upsilon,$$

i.e., we optimize E with F as the objective function (*optimization criterion*). Note that the effect of protection is evaluated from High's perspective, while the database functionality is evaluated from Low's view.

5 Modification Control

In theory, the optimal set of attribute values for modification can be determined by exhaustively evaluating every possible batch of attribute values of non-confidential data and selecting the batch that scores the highest with respect to a given optimization criterion. Such an exhaustive search is impractical when the volume of the data is large. Instead of evaluating each attribute in turn, we prioritize the attributes and evaluate the one that yields the highest *threat*. During modification, we visit the attribute that has the largest number of sensitive data records and the lowest classification error - the highest inference threat. (Prioritization needs to be carried out for each run of modification.) Let the total number of confidential attribute values be denoted as S, and the classification error of the test data with respect to the jth attribute be Cr_j. We select from all the M attributes the one that maximizes the product of the number of associated confidential data records and the inverse of the classification error

$$MAX_{j=1}^{M}\ (1 - Cr_j)(\frac{TE_j}{S})$$

where TE_j is the number test data associated with the jth attribute. Handling attributes that are of high inference threat first allows us to achieve the necessary level of protection more effectively.

6 Example

Table 2 is a small sample taken from a Submarine Design database[1] and represents the initial Low view. In-

[1]The Submarine Design database has 98 records and 9 attributes and was collected in the Jane's Naval Weapon Systems.

Table 2: initial Low database

name	*diesels*	*range*	*depth*
Agosta	low	short	medium
Foxtrot	medium	medium	medium
Seawolf	high	long	deep
S. Cruz	?	medium	medium
Preveze	low	?	medium

formation of "diesels of S. Cruz" and "range of Preveze" is assumed to be sensitive. From the available information (D), one can infer these two pieces of sensitive data with probability 1, i.e., Pr("diesels of S. Cruz" = medium | D) = 1 and Pr("range of Preveze" = short | D) = 1. The inference is excessive. After downgrading, the modified Low view is shown in Table 3, where modified attribute values (i.e., "?"s) are in bold-face. The probabilities of these two pieces of sensitive data become Pr("diesels of S. Cruz" = medium | D) = 0.33 and Pr("range of Preveze" = short | D) = 0.33, indicating equal likelihood, and the result is desirable.

Table 3: modified Low database

name	*diesels*	*range*	*depth*
Agosta	low	short	medium
Foxtrot	medium	medium	medium
Seawolf	high	long	deep
S. Cruz	?	**?**	medium
Preveze	**?**	?	medium

7 Restoration Attacks

If an adversary knows the strategy of inference prevention, then (s)he may be able to restore the modified attribute values (referred to as the *restoration attack*). In this case, the sensitive data are not correctly protected.

As discussed, sensitive data associated with an attribute are deemed as the classes of the test data when this attribute is posted as the class_label. For a decision tree, the root node (attribute) are more likely to be selected for modification than other nodes (attributes), because the root node is the most informative attribute (in terms of the Shannon's entropy measure) to the class_lable. Among many, we consider one type of restoration attack in which an adversary computes the most informative attribute w.r.t a class_label, posts it as the new class_lable, and estimates those associated hidden values. For example, consider the "voting" data ([4]). In this data set, the original class_lable is "party" and the corresponding most informative attribute is "physician fee freeze". It can be shown that the previously hidden attribute values (e.g., "physician fee freeze") are restored by using some other attributes (e.g., "El Salvador aid"). ("El Salvador aid" is the most informative attribute to "physician fee freeze".) To prevent the possible restoration of modified values, we repeat the process of attribute value hiding by making previously modified non-sensitive data sensitive until the restoration risk drops below a specified threshold. (Of course, this threshold is incorporated in F.) The need of repeated hiding is referred to as the *ramification* problem of data inference ([2]).

8 Future Work

We will study the effects of inference prevention based on different modification methods, investigate different types of statistical attack, and extends the current model to distributed environments.

References

[1] Chang, L. & Moskowitz, I. (1998) "Parsimonious Downgrading and Decision Trees Applied to the Inference Problem," NSP Workshop, pp. 82-89.

[2] Chang, L & Moskowitz, I. (2000) "An Integrated Framework for Database Inference and Privacy Protection," *Data And Applications Security*, (eds. Thuraisingham, van de Riet, Dittrich & Tari), Kluwer, pp. 161-172.

[3] P. Doyle, J. Lane, L. Zayatz, and J. Theeuwes. *Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies.* Elsevier Science, 2001.

[4] http://kdd.ics.uci.edu/ .

[5] Quinlan, R. (1992) *C4.5*, Morgan Kaufmann.