# A Framework for Privacy Preserving Classification in Data Mining

**Md. Zahidul Islam and Ljiljana Brankovic**

School of Electrical Engineering and Computer Science
The University of Newcastle
Callaghan, NSW 2308, Australia

zahid@cs.newcastle.edu.au, lbrankov@cs.newcastle.edu.au

## Abstract

Nowadays organizations all over the world are dependent on mining gigantic datasets. These datasets typically contain delicate individual information, which inevitably gets exposed to different parties. Consequently privacy issues are constantly under the limelight and the public dissatisfaction may well threaten the exercise of data mining and all its benefits. It is thus of great importance to develop adequate security techniques for protecting confidentiality of individual values used for data mining.

In the last 30 years several techniques have been proposed in the context of statistical databases. It was noticed early on that non-careful noise addition introduces biases to statistical parameters, including means, variances and covariances, and sophisticated techniques that avoid biases were developed. However, when these techniques are applied in the context of data mining, they do not appear to be bias-free. Wilson and Rosen (2002) suggest the existence of Type Data Mining (DM) bias that relates to the loss of underlying patters in the database and cannot be eliminated by preserving simple statistical parameters. In this paper we propose a noise addition framework specifically tailored towards the classification task in data mining. It builds upon some previous techniques that introduce noise to the class and the so-called innocent attributes. Our framework extends these techniques to the influential attributes; additionally, it caters for the preservation of the variances and covariances, along with patterns, thus making the perturbed dataset useful for both statistical and data mining purposes. Our preliminary experimental results indicate that data patterns are highly preserved suggesting the non-existence of DM bias.

Keywords: Data mining, Statistical database, Data security, Privacy, Noise addition

## 1 Introduction

The advances in information processing technology and the storage capacity have established data mining as a widely accepted technique for various organizations. The benefits of data mining include but are not restricted to improvement in diagnosis of diseases, composition of drugs tailored towards individual's genetic structure (US Department of Energy 2001), and automatic aid for deciding whether a loan application should be accepted or not. In modern days organizations are extremely dependent on data mining in their every day activities and paybacks include providing better service, achieving greater profit, and better decision-making. For these purposes organizations collect huge amount of data on which they apply data mining techniques. For example, business organizations collect data about the consumers for marketing purposes and improving business strategies, medical organizations collect medical records for better treatment and medical research, and national security agencies maintain criminal records for security purpose.

Typically, these data are collected with the consent of the data subjects and the collector provides some assurance that the privacy of individual data will be protected. However, the secondary use of collected data is also very common. Secondary use is any use for which data were not collected initially. Additionally, some organizations sell the collected data to other organizations, which use these data for their own purposes. Thus the data get exposed to a number of parties including collectors, owners, users and miners.

Individual records are often considered to be private and sensitive. For example, detailed credit card records can disclose personal lifestyle with reasonable accuracy. Any misuse of such personal data can be uncomfortable for the individuals and can occasionally cause a serious damage to them. Consequently, public concern about the personal information is growing which forces governments and law enforcing agencies to introduce and implement new privacy protecting laws and regulations. An example of such laws is the US Executive Order (2000) that protects federal employees from being discriminated, on the basis of protected genetic information, for employment purposes (Clinton 2000). It is not unlikely that stricter privacy laws will be introduced in the future. On the other hand, without such laws individuals may become hesitant to share their personal information. Both scenarios may make the data collection difficult and hence may deprive the organizations from the benefits of data mining resulting in inferior quality of services provided to the public. Such prospects equally concern collectors and owners of data, as well as researchers. A possible solution to this problem is to provide security techniques that would insure privacy of individual records while at the same time enable data mining and statistical analysis. In recent years several security techniques in statistical database and data mining have been proposed.

It is deemed that the confidentiality of individual records can be maintained and at the same time the usefulness of

data for data mining can also be preserved (Agrawal and Srikant 2000, Estivil-Castro and Brankovic 1999, Islam and Brankovic 2003). We observe that removing the names and other identifiers may not guarantee the confidentiality of individual records as it is often the case that a particular record can be uniquely identified from the combination of other attributes. Hence we need more sophisticated protection techniques. The datasets used for data mining purposes do not necessarily need to contain 100% accurate data. In fact, that is almost never the case, due to the existence of natural noise in datasets. In the context of data mining it is important to maintain the patterns in the dataset. Additionally, maintenance of statistical parameters, namely means, variances and covariances of attributes is important in the context of statistical databases. However, the maintenance of only statistical parameters does not necessarily preserve the patterns in a dataset. To illustrate this, let us consider two attributes, *A* and *B*. The correlation of these two attributes over the whole population space may not be significantly high but they may have a very high correlation in a specific part of the dataset, say when a value of *A* is within a specific range. For example, A could be "Age" and B could be "Height". While these two attributes are highly correlated when Age is in the range 0 – 10, their correlation is very low in the range 20+. This high correlation between the attributes in a particular part of the dataset can be seen as a pattern of the dataset. The maintenance of the correlations among the attributes over the whole population space might not preserve the patterns.

In this paper we concentrate on classification and pattern preservation of a dataset while adding noise to it. At the same time we are maintaining the statistical parameters in order to provide the usefulness of the dataset for statistical purposes. We propose a noise addition framework, which uses a few previous techniques (Estivil-Castro and Brankovic 1999, Islam and Brankovic 2003, Muralidhar, Parsa and Sarathy 1999) and incorporates them with new technique introduced in this paper. Our initial experimental results, presented in this paper, indicate that the proposed framework preserves the pattern of datasets while at the same time it provides high privacy by perturbing the data of a dataset. This framework perturbs both confidential and non-confidential attributes. Perturbation of non-confidential attributes is very valuable in providing a higher level of security, as it makes it difficult for an intruder to uniquely identify a record and thus compromise its confidential values.

## 2 Previous Work

Many privacy techniques have been proposed in the context of statistical databases (see, for example, Adam and Wortmann 1989, Muralidhar, Parsa and Sarathy 1999, Tendick and Norman 1987, Tendick and Matloff 1994) while few privacy techniques have been proposed in the context of data mining (see, for example, Agrawal and Srikant 2000, Du and Zhan 2002, Lindell and Pinkas 2000, Oliviera and Zaane 2002). An excellent survey, on the existing privacy methods for statistical databases, has been presented in (Adam and Wortmann 1999). These methods have been categorised in three main groups, based on the approaches they take, such as query restriction, data perturbation, and output perturbation. Among these methods data perturbation method is the most straightforward to implement. We only need to perturb the data. Then we can use any existing software (eg. DBMS) to access the data without any further restrictions on processing. That is not the case with query restriction and output perturbation. Thus, we feel that this method is the most suitable for data mining applications, and to our knowledge no other methods have been investigated in this context.

The simplest version of additive data perturbation was proposed by Kim (1986). This method suffers from bias related to the variances (Type A), bias related to the relationship between confidential attributes (Type B) and bias related to the relationship between confidential and non-confidential attributes (Type C). Several years later, Tendick and Matloff (1994) presented a modified version of random data perturbation, which is free of Type A and Type B bias. In 1999, Muralidhar, Parsa and Sarathy (1999) proposed a new data perturbation method called *General Additive Data Perturbation (GADP)* method. For the database with a multivariate normal distribution, the *GADP* method is said to be the most secure and the least bias prone. This method is structured to be free of the so-called Type "A", Type "B", Type "C" and Type "D" bias.

In 2002, Wilson and Rosen (2002) compared *GADP* method with a naïve noise addition method called *Simple Additive Data Perturbation (SADP)* method in the context of data mining. They suggested the existence of a new bias called Type Data Mining (DM) bias and thus attempted to show that *GADP* method is not bias free in the context of data mining.

Estivil-Castro and Brankovic (1999) proposed a data perturbation technique by adding noise to the class attribute. The technique emphasised the pattern preservation instead of obtaining unbiased statistical parameters.

The majority of the previous studies evaluate the quality of the perturbed datasets by measuring the predictive accuracy of the classifiers built on the perturbed datasets (Kohavi 1996, Lim, Loh and Shih 2000, Wilson and Rosen 2002). However, in this paper we follow our previous directions (Estivil-Castro and Brankovic 1999, Islam and Brankovic 2003, Islam, Barnaghi and Brankovic 2003) and evaluate the quality of the perturbed datasets by comparing the decision trees and the logic rules associated with the perturbed and the original datasets. Our previous work (Islam, Barnaghi and Brankovic 2003) indicates that the preservation of logic rules is highly correlated with the accuracy of neural network classifiers, which are known to be very sensitive to noise.

## 3 The Framework

In this section we present a noise addition framework for datasets that contain several numerical attributes and a single categorical attribute (class). An example of such dataset is the *Wisconsin Breast Cancer (WBC)* dataset available from the UCI Machine Learning Repository at http://www.ics.uci.edu/~mlearn/MLRepository.html.

We first give brief introduction to *WBC* dataset, as we shall refer to it in examples throughout this section.

The *WBC* dataset has 10 numerical non-class attributes and one categorical class attribute. The class attribute has two categorical values, "2" and "4". Out of 10 numerical attributes one is the Record ID. The 9 numerical attributes (excluding the record ID) are Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli and Mitoses. Each of these attributes draws an integer value from the range 1 to 10 inclusive.

We used Quinlan's C5 decision tree builder on 349 cases of the *WBC* dataset, and we obtained a decision tree shown in Figure 1. Ellipses represent internal nodes and circles represent the leaves.

We next present a framework for adding noise to all the attributes in a dataset, including the class, in such a way that the patterns discovered by the decision tree built on the original dataset are preserved. Additionally, our framework can be extended so as to preserve the correlation among the attributes. This extension makes the framework applicable to a wider range of datasets, both those to be used for classification and those used for statistical analysis. Our preliminary experimental results indicate that the patterns are very well preserved (see Section 4).

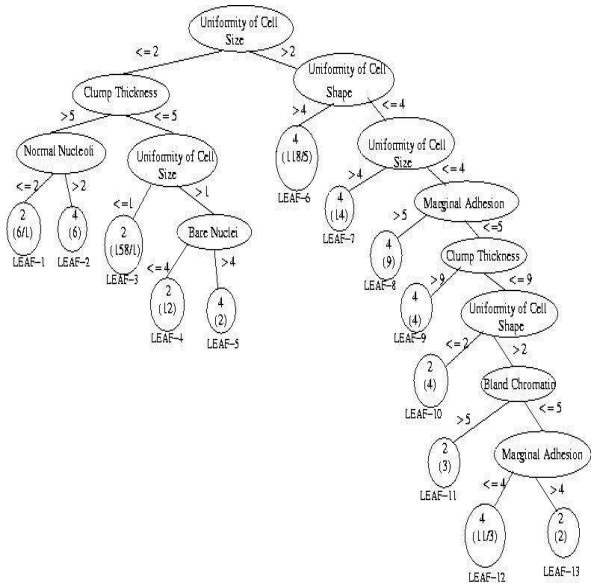


**Figure 1.** The decision tree obtained from 349 cases of the unperturbed WBC dataset.

The framework engages three steps as described below.

**Step 1:** Add noise to *Leaf Influential Attributes* (*LINA*s), of each leaf of the decision tree obtained from the unperturbed dataset, by *Leaf Influential Attribute Perturbation Technique* (*LINAPT*).

**Step 2:** Add noise to *Leaf Innocent Attributes (LIA*s*)*, of each leaf of the decision tree obtained from the unperturbed dataset, by *Leaf Innocent Attribute Perturbation Technique (LIAPT).*

**Step 3:** Add noise to class attribute, for each of the heterogeneous leaves, by *Random Perturbation Technique (RPT).*

In order to describe the details of the three framework steps, we first need to introduce some assumptions and definitions.

First of all, we note that we add noise to all of the attributes regardless of whether they are considered to be confidential or non confidential. Hence we consider all the attributes, including the class attribute, sensitive. The rational behind this assumption is that considering some of the attributes to be non-sensitive and not adding any noise to those attributes can facilitate identification of a particular record by an intruder. If the identification is possible then the intruder can learn the values of all other attributes of that record. Thus in that case it would be absolutely necessary to add a significant amount of noise to all confidential attributes in order to protect them, and adding a great deal of noise would affect the patterns in the dataset and to some extent the predictive accuracy of the classifier built on the perturbed dataset. On the other hand, if the identification of individual records is made difficult by adding noise to all the attributes, or in other words if all the attributes are considered to be sensitive, then the same level of privacy can be achieved by adding less noise to confidential attributes. By spreading the noise in such a way, we may better preserve the patterns. This is particularly important in the context of "Influential Attributes", where we can only add a limited noise as described later in this section.

Secondly, we split the non-class attributes into two divisions, namely *Leaf Innocent Attributes* (*LIA*s) and *Leaf Influential Attributes* (*LINA*s) for each leaf. In a decision tree, if an attribute is not tested in any of the nodes on the path between the root and a leaf, then the attribute is called *Leaf Innocent Attribute (LIA)* for that particular leaf (Islam and Brankovic 2003). Thus, every leaf of a decision tree maintains a set of *LIA*s. For example, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin and Mitoses are the *Leaf Innocent Attributes (LIA*s*)* for LEAF-1 and LEAF-2 of the decision tree shown in Figure 1.

On the contrary, the attribute that is tested at least once on the path between the root and a leaf is called *Leaf Influential Attribute (LINA)* for that particular leaf. Every leaf of a decision tree upholds a set of *LINA*s. For example, Clump Thickness, Uniformity of Cell Size and Normal Nucleoli are the *Leaf Influential Attributes (LINA*s*)* for LEAF-1 and LEAF-2.

Finally, we define a domain of an attribute to be a set of legal values that the attribute can take (Date 2000). For example, the domain of the 9 numerical attributes in *WBC* database is [1,10].

Now we are ready to explain in details the three steps of the framework.

In the **Step 1** we introduce a new technique for perturbing the *Leaf Influential Attributes* (*LINA*s). We call this technique *Leaf Influential Attribute Perturbation*

*Technique* (*LINAPT*). For each leaf of the decision tree *LINAPT* first identifies the *LINA*s of that leaf. *LINAPT* then adds noise to each record of that leaf, and only to *Leaf Influential Attributes*. Let *A* be a *LINA*. After *LINAPT* adds noise to *A*, the perturbed value *A'* of the same attribute will be $A' = A + \varepsilon$, where $\varepsilon$ is a discrete noise with mean μ and variance $\sigma^2$. The distribution of the noise can be chosen to suit a particular application.

An important characteristic of *LINAPT* is that all perturbed values remain within the range defined by the conditional values in *LINA*s for that particular leaf. For example, LEAF-1 of Figure 1 has three *LINA*s namely Uniformity of Cell Size, Clump Thickness and Normal Nucleoli. The range of Uniformity of Cell Size's defined by the conditional value for LEAF-1 is 1 to 2 inclusive (see Figure 1). Thus *LINAPT* adds noise to this attribute for the cases belong to the LEAF-1 in such a way so that the perturbed value of the attribute remains within the range 1 to 2. Similarly, the range of Uniformity of Cell Size for LEAF-10 is 3 to 4 inclusive, while the range of the Uniformity of Cell Shape is 1 to 2 inclusive (see Figure 1). *LINAPT* perturbs all the *LINA*s of each leaf in the same way.

In the **Step 2** we perturb the *LIA*s according to the *Leaf Innocent Attribute Perturbation Technique (LIAPT)*, proposed in (Islam and Brankovic 2003). *LIAPT* is used to perturb the *LIA*s within each leaf. *LIA*s are not influential in the sense that they do not play any role in prediction of the class attribute for that particular leaf. In other words, they do not appear in any logic rule that is related to the leaf. Hence they are not considered important for the pattern preservation of the dataset.

*LIAPT* first detects the *LIA*s from the original decision tree and then adds noise to each *LIA*. Let *B* be a *LIA*. After *LIAPT* adds noise to *B*, the value of the same attribute will be $B' = B + \varepsilon$, where $\varepsilon$ is a discrete noise with mean μ and variance $\sigma^2$. Again the distribution of the noise can be chosen to suit particular application.

The *LIAPT* was first introduced in (Islam and Brankovic 2003). In the same paper we presented some experimental results to evaluate the technique. We only perturbed *LIA*s of the heterogeneous leaves of *WBC* dataset. We used an approximation of a normal distribution with mean μ=0 and standard deviation $\sigma$=27.6% of the attribute value. We then compared the decision tree produced from the perturbed dataset to the decision tree produced from unperturbed dataset. We carried out the same experiment 10 times. In 8 out of 10 experiments the trees obtained from the perturbed datasets were very similar to the tree obtained from the unperturbed dataset. Only in two out of 10 experiments we obtained decision trees which are slightly different from the original tree but nevertheless they hold the same root and some logic rules.

In this paper we extend the experiment with *LIAPT* by perturbing *LIA*s in both homogeneous and heterogeneous leaves by *LIAPT*. We present our experimental results in the next section.

In **Step 3** we use a perturbation technique to preserve the confidentiality of the class attribute. Three class attribute perturbation techniques, namely *Random Perturbation Technique (RPT)*, *Probabilistic Perturbation Technique (PPT)* and *All Leaves Probabilistic Perturbation Technique (ALPT)* were proposed in (Islam and Brankovic 2003). Out of these three class attribute perturbation techniques we found *Random Perturbation Technique (RPT)* to be the best (Islam and Brankovic 2003, Islam, Barnaghi and Brankovic 2003) and hence our framework uses *RPT*. In *RPT* we first focus on a heterogeneous leaf of a decision tree and find the number of cases with the minority class in that heterogeneous leaf. We denote this number as *n*. We then convert all *n* cases of the minority class to the majority class. Then we randomly select *n* cases belong to the same leaf and convert the class of those cases from majority to minority. We continue the perturbation of class attribute for all heterogeneous leaves of the decision tree. For example, LEAF-1, LEAF-3, LEAF-6 and LEAF-12 are heterogeneous leaves of the decision tree shown in Figure 1. We observe that the minority class for LEAF-1 is "4" and there is one case with minority class in the leaf. In *RPT* we first convert the class of that case from "4" to "2". Then select any one case randomly out of all 6 cases belong to the leaf and convert the class of the selected case from "2" to "4". Similar perturbation is performed for the remaining heterogeneous leaves of the decision tree.

In (Islam and Brankovic 2003), we presented experimental results on *RPT*. We produced a decision tree from 300 cases of Boston Housing Price dataset, which is available from UCI Machine Learning Repository at http://www.ics.uci.edu/~mlearn/MLReposi_tory.html. We then perturbed the dataset by *RPT* technique 5 times, and each time we produced a decision tree from the perturbed dataset. Hence we got 5 decision trees from 5 perturbed datasets, and we compared them with the original decision tree. We found that the logic rules of the perturbed decision trees and the logic rules of the original decision tree were similar. Out of four attributes tested in the original decision tree, three were tested in every decision tree. The conditional values of these attributes were similar in decision trees obtained from original and perturbed datasets.

We now propose an extension of our framework in order to maintain the statistical parameters of a dataset, as well as patterns.

***For each leaf, DO:***

> **Step 1:** Add noise to *Leaf Influential Attributes* (*LINA*s), and *Leaf Innocent Attributes* (*LIA*s), using the *GADP* technique (Muralidhar, Parsa and Sarathy 1999), where the domain of each *LINA* is bounded by the conditional values for that leaf.
>
> **Step 2:** Add noise to class attribute, for each of the heterogeneous leaves, by *Random Perturbation Technique (RPT).*

***END DO***

In **Step 1** we consider the collection of records that belong to the leaf under consideration to be a dataset in

its own right. The domains of *LIA*s remain the same as their domains in the original dataset. However, the domains of the *LINA*s are defined by the conditional values for the leaf. Thus, after the perturbation is completed the values of *LINA*s will remain in the range defined by the conditional values.

**Step 2** is straightforward and remains the same as the Step 3 of the original framework.

The extended framework effectively partitions the dataset into partitions defined by the leaves of the decision tree built from the unperturbed dataset. The *GADP* method is then applied to each leaf separately. This guarantees the preservation of the correlations among all the attributes, as well as absence of any bias of types A, B, C or D. Thus the correlations in the dataset as a whole will also be preserved and the above mentioned biases will not occur. The advantage of adding noise leaf by leaf is in the preservation of all the patterns discovered by the original tree. The trade off that we must bear is in certain decrease of the security, as an intruder will have tight bounds on the original values in *LINA*s. However, there are no such bounds on *LIA*s.

## 4 Experimental Results

In this section we present results of our experiments on Leaf Influential Attribute Perturbation Technique *(LINAPT)* and *Leaf Innocent Attribute Perturbation Technique (LIAPT).*

### 4.1 Experimental Results on *LIANPT*

In our experiments on *LINAPT* we used noise with the mean $\mu = 0$ and a standard deviation $\sigma$ which depends on the new domains of *LINAs*. For larger values of domains of *LINA*s $\sigma$ can be approximated as 27.6% of the domain. In fact, we used a probability distribution of the noise as shown in Figure 2. Horizontal axis represents noise as a percentage of the range of the attribute value defined by the conditional value for a leaf of a decision tree. Vertical axis represents the probability of adding the corresponding noise. The $\sigma$ is decreasing with a decrease in the domain due to the rounding factor.

We used the 349 cases of *WBC* dataset for our experiments. We first produced a decision tree, shown in Figure 1, from the original 349 cases of *WBC* dataset. Then we applied *LINAPT* on each leaf of the original decision tree (Figure 1). Thus we produced a perturbed dataset and then we obtained a decision tree from it. We repeated the experiment 15 times. Hence, we produced 15 perturbed datasets and therefore 15 perturbed decision trees.

We compared the logic rules of the decision trees obtained from perturbed datasets with the logic rules of the decision tree obtained from the original dataset. Although decision trees are generally known as instable to noise (Li 2001), in 12 out of 15 experiments we find that the decision tree obtained from the perturbed dataset is exactly same or almost the same as the original decision tree. More precisely, in 7 out of 15 experiments we obtained trees which are identical to the original tree. The trees obtained in the further 3 experiments were almost the same as the original tree. The only difference was in one node, which is at the very bottom of the tree. This node only affects 12 to 13 cases out of 349. The remaining part of these 3 trees is exactly the same as in the original tree (see Figures 1 and 3 for illustration). Thus, all logic rules associated with these trees are exactly the same as the logic rules associated with the original tree, except for the two rules defined by the node that is different.

Further two trees are very similar to the original tree. They both test the same 7 attributes as the original tree, and have 13 nodes each (the original tree has 12 nodes). Out of these nodes, 10 nodes are exactly the same as in the original tree, in the sense that they test the same attributes with the same conditional values and produce leaves with the same number of cases as the original tree. Ten out of 14 logic rules associated with these 2 trees are the same as in the original tree, and they cover 329 out of 349 cases.

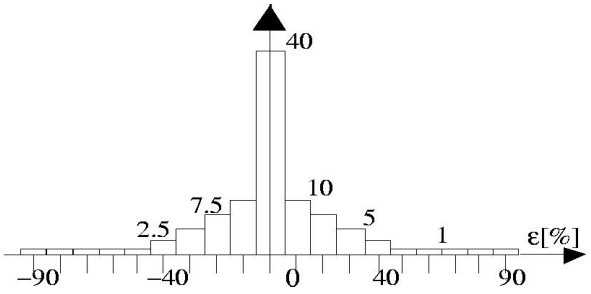


**Figure 2.** Probability distribution of the noise added to *LINA*s.

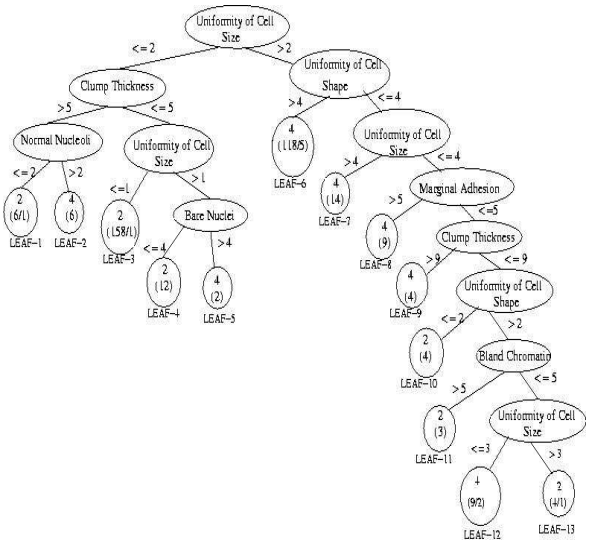


**Figure 3.** A decision tree obtained from a dataset perturbed by *LINAPT*. This decision tree is very similar to the decision tree obtained from unperturbed dataset.

The remaining 3 trees are slightly different from the original tree. They have between 8 and 13 nodes and between 9 and 14 leaves, while the original tree has 12 nodes and 13 leaves. However, around half of the logic rules associated with the original tree also appear in all of the 3 trees and these logic rules cover 316 out of 349

cases in each tree. All of these trees test at least 6 out of 7 attributes tested in the original tree. Figure 4 shows one of these 3 trees.

These experimental results indicate that *LINAPT* is excellent at preserving the patterns in the dataset. To fully appreciate this, a reader needs to remember that decision trees are very sensitive to the noise. Our previous experiments (Islam and Brankovic 2003) show that when the noise is added randomly to the records of the dataset, the decision trees typically suffer major changes in respect to number of nodes and logic rules associated with the tree.

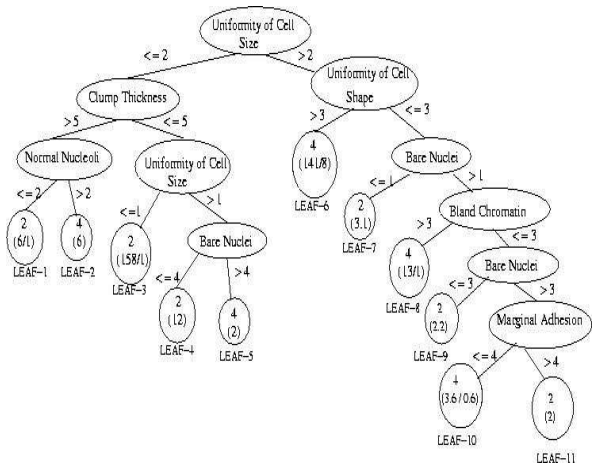


**Figure 4.** A decision tree that is obtained from a dataset perturbed by *LINAPT*. This tree is considered not to be very similar to the original tree.

### 4.2 Experimental Results on *LIAPT*

In our experiments on *LIAPT* we used mean $\mu = 0$ and a standard deviation $\sigma =$ 27.6% of the range of the attribute. We used a similar probability distribution of the noise shown in Figure 2. We used the same *LIAPT* in (Islam and Brankovic 2003), but there we perturbed only the *LIA*s of all heterogeneous leaves of the decision tree. However, in the experiments presented in this paper we perturbed the *LIA*s of each and every leaf of the original decision tree. Thus we produced a perturbed dataset and a perturbed decision tree. We repeated the experiment 10 times. In 7 out of 10 experiments the decision trees were exactly same as the original decision tree. In the remaining 3 experiments we obtained trees that are different from the original tree. However, they still preserve many logic rules of the original decision tree.

## 5 Conclusion

In this paper we propose a noise addition framework for protecting privacy of sensitive information used for data mining purposes. The framework does not distinguish between confidential and non-confidential attributes but rather adds noise to all of them. Adding noise to non-confidential attributes contributes to the overall security by preventing unique identification of records by an intruder. The framework can be extended so that it incorporates *GADP* method proposed by Muralidhar et al. This method is known to be secure while at the same time it does not introduce any of the four statistical biases (Muralidhar, Parsa and Sarathy 1999, Wilson and Rosen 2002). Our experiments indicate that when use within our framework *GADP* may also be considered free of bias Type DM.

## 6 References


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

Agrawal, R. and Srikant, R. (2000): Privacy-preserving Data Mining, In *Proceedings of the 2000 ACM SIGMOD Conference on Management of Data*, Dallas, Tx.

Clinton, W.J., Executive Order 13145, 2000, http://www.dol.gov/oasam/regs/statutes/eo13145.htm. Accessed 7 Oct 2003

Date, C., J.(2000): *An Introduction to database Systems*, 7th edition, Addison-Wesley.

Du, W. and Zhan, Z. (2002): Building Decision Tree Classifier on Private Data. In Proc. of IEEE *Workshop on Privacy, Security and Data Mining, at the ICDM 02*, Conferences in Research and Practice in Information Technology, **14**, Clifton, C and Estivill-Castro, V, eds.

Estivil-Castro, V. and Brankovic, L. (1999): Data Swapping: Balan-cing Privacy Against Precision in Mining for Logic Rules, M. Mohania and A.M. Tjoa, eds., *Data Warehousing and Know-ledge Discovery DaWaK'99, LNCS 1676*, 389-398.

Islam, M. Z. and Brankovic, L. (2003): Noise Addition for Protecting Privacy in Data Mining, *Proceedings of The 6th Engineering Mathematics and Applications Conference (EMAC2003)*, Sydney, 85-90.

Islam, M. Z., Barnaghi, P. M. and Brankovic, L (2003): Measuring Data Quality: Predictive Accuracy vs. Similarity of Decision Trees, Accepted for publication in *Proceedings of the 6th International Conference on Computer and Information Technology (ICCIT-2003), Jahangirnagar Univ., Bangladesh*.

Kim, J. (1986):A Method for Limiting Disclosure in Microdata Based on Random Noise and Transformation, In *Proceedings of the American Statistical Association on Survey Research Methods, American Statistical Association*, Washington, DC, 370-374.

Kohavi, R. (1996): Scaling Up the Accuracy of Naïve-Bayes Classifiers: a Decision-Tree Hybrid, In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*.

Li, R.-H. (2001): *Instability of Decision Tree Classification Algorithms*, PhD Thesis, University of Illinois at Urbana-Champaign.

Lim, T.-S., Loh, W.-Y. and Shih, Y.–S. (2000): A Comparison of Predictive Accuracy, Complexity and Training Time of Thirty Three Old and New Classification Algorithm, *Machine Learning Journal*, **40**, 203-228.

Lindell, Y. and Pinkas, B. (2000): Privacy Preserving Data Mining, M. Bellare (Ed.): *Proceedings of the Advances in Cryptology - CRYPTO 2000*, LNCS 1880.

Muralidhar, K., Parsa, R. and Sarathy, R. (1999): A General Additive Data Perturbation Method for Database Security, *Management Science*, **45** (10), 1399-1415.

Oliveira, S.R.M. and Zaane, O.R. (2002): Foundations for an Access Control Model for Privacy Preservation in Multi-Relational Association Rule Mining, *Workshop on Privacy, Security and Data Mining*, at the ICDM 02, Conferences in Research and Practice in Information Technology, **14**, Clifton, C. and Estivill-Castro, V., eds.

Tendick, P., and Norman, N.S. (1987): Recent Result On The Noise Addition Method For Database Security, *In Proceedings of the 1987 Joint Meetings, American Statistical Association / Institute of Mathematical Statistics. ASA/IMA*, Washington, DC.

Tendick, P. and Matloff, N. (1994): A Modified Random Perturbation Method for Database Security, *ACM Transaction on Database Systems*, **19** (1), 47-63.

US Department of Energy, Human Genome Program, Genomics and Its Impact on Medicine and Society: A 2001 Primer, http://www.ornl.gov/hgmis/publicat/primer2001/.

Wilson, R. L. and Rosen, P. A. (2002): The Impact of Data Perturbation Techniques on Data Mining, *In Proceedings of the 33rd Annual Meeting of DSI*.