Data Mining Techniques in Fraud Detection

Rekha Bhowmik
University of Texas at Dallas
rekha.bhowmik@utdallas.edu

ABSTRACT

The paper presents application of data mining techniques to fraud analysis. We present some classification and prediction data mining techniques which we consider important to handle fraud detection. There exist a number of data mining algorithms and we present statistics-based algorithm, decision tree-based algorithm and rule-based algorithm. We present Bayesian classification model to detect fraud in automobile insurance. Naïve Bayesian visualization is selected to analyze and interpret the classifier predictions. We illustrate how ROC curves can be deployed for model assessment in order to provide a more intuitive analysis of the models.

Keywords: Data Mining, Decision Tree, Bayesian Network, ROC Curve, Confusion Matrix

1. INTRODUCTION

Data mining refers to extracting or "mining" knowledge from large amount of data. There are a number of data mining techniques like clustering, neural networks, regression, multiple predictive models. Here, we discuss only few techniques of data mining which would be considered important to handle fraud detection. They are i) Bayesian network, for classifying risk group, and ii) Decision tree, for creating descriptive model of each risk group.

Data Mining is associated with (a) supervised learning based on training data of known fraud and legitimate cases and (b) *unsupervised learning* with data that are not labeled to be fraud or legitimate. Bedford's law can be interpreted as an example of unsupervised learning (Bolton et al. 2002). The direct application of these methods to forensic accounting is limited due to almost complete nonexistence of large sets of fraud training data (Bolton et al. 2002; Jensen, 1997).

Insurance fraud, credit card fraud, telecommunications fraud, and check forgery are some of the main types of fraud. Insurance fraud is common in automobile, travel. The Uniform Suspected Insurance Fraud Reporting Form, adopted by the NAIC Antifraud Task Force 2003, replaced the prior Task Force form. This form standardizes insurance fraud data for the insurance industry and makes it easier to report and track. Fraud detection involves three types of offenders (Baldock, 1997): i) Criminal offenders, ii) organized criminal offenders who are responsible for major fraud, and iii) offenders who commit fraud (called soft

fraud) when suffering from financial hardship. Soft fraud is the hardest to lessen because the cost for each suspected incident is usually higher than the cost of the fraud (National White Collar Crime Center, 2003). Types i) and ii) offenders, called hard fraud, avoid anti-fraud measures (Sparrow, 2002).

We present data mining techniques which are most appropriate for fraud analysis. We present automobile insurance example. Three data mining techniques used for fraud analysis are: i) Bayesian network, ii) Decision tree, and iii) backpropagation. Bayesian network is the technique used for classification task. Classification, given a set of predefined categorical classes, determines which of these classes a specific data belongs to. Decision trees are used to create descriptive models. Descriptive models are created to describe the characteristics of fault.

The remainder of this paper is organized as follows. In Section 2, we present the existing fraud detection systems and techniques. Section 3 the three classification algorithms and application. Section 4 presents the data. Finally, in section 5, we discuss the important features of our work and further work.

2. EXISTING FRAUD DETECTION SYSTEMS

A fuzzy logic system (Altrock et al. 1995) incorporated the actual fraud evaluation policy using optimum threshold values. The result showed the chances of fraud and the reasons why an insurance claim is fraudulent. This system predicted slightly better results than the auditors. Another logic system (Cox et al. 1995) used two approaches to imitate the reasoning of fraud experts, i) the discovery model, uses an unsupervised neural network to find the relationships in data and to find clusters, then patterns within the clusters are identified, and ii) the fuzzy anomaly detection model, which used Wang-Mendel algorithm to find how health care providers committed fraud against insurance companies. The EFD system (Major et al. 1995) integrated the expert knowledge with statistical information to identify providers whose behavior did not fit the rule.

The hot spots methodology (Williams et al. 1997**)** performed a three step process: i) k-means clustering algorithm for cluster detection is used because the other clustering algorithms tend to be computationally expensive where the datasets are very large, ii) C4.5 algorithm, the resulting decision tree can be converted to a rule set and pruned, and iii) visualization tools for rule evaluation, building statistical summaries of the entities associated with each rule. (Williams, 1999**)** extended the hot spots methodology to use genetic algorithms to generate and explore the rules.

The credit fraud model (Groth et al. 1998) suggested a classification technique with fraud/legal attribute, and a clustering followed by a classification technique with no fraud/legal attribute. Kohonen's Self-Organizing Feature Map (Brockett et al. 1998) was used to categorize automobile injury claims depending on the size of fraud suspicion. The validity of the Feature Map was then evaluated using a back propagation algorithm and feed forward neural networks. Result showed that the

method was more reliable and consistent compared to the fraud assessment.

Classification techniques have proved to be very effective in fraud detection (He et al. 1998; Chen et al. 1999) and therefore, can be applied to categorize crime data. The distributed data mining model (Chen et al. 1999) uses a realistic cost model to evaluate C4.5, CART, and naïve Bayesian classification models. The method was applied to credit card transactions. The neural data mining approach (Brause et al. 1999) uses rule-based association rules to mine symbolic data and Radial Basis Function neural network to mine analog data. The approach discusses the importance of use of non-numeric data in fraud detection. It was found that the results of association rules increased the predictive accuracy.

SAS Enterprise Miner Software (SAS e-intelligence, 2000) depends on association rules, cluster detection and classification techniques to detect fraudulent claims. The Bayesian Belief Network (BBN) and Artificial Neural Network (ANN) study used the STAGE algorithm for BBN in fraud detection and backpropagation for ANN (Maes et al. 2002). STAGE repeatedly alternates between two stages of search: running the original search method on objective function, and running hill-climbing to optimize the value function. The result shows that BBNs were much faster to train, but were slower when applied to new instances. FraudFocus Software (Magnify, 2002) automatically scores all claims. The scores are sorted in descending order of fraud potential and generate descriptive rules for fraudulent claims. FairIsaac(Weatherford et al. 2002) recommended backpropagation neural networks for fraudulent credit card use. The ASPECT group (Weatherford et al. 2002) focused on neural networks to train current user profiles and user profiles histories. A caller's current profile and the profile history are compared to find probable fraud. (Cahill et al. 2002) build on the adaptive fraud detection framework (Fawcett et al. 1997) by applying an event-driven approach of assigning fraud scores to detect fraud. The (Cahill et al. 2002) framework can also detect types of fraud using rules. This framework has been used in both wireless and wired fraud detection systems. (Ormerod el al. 2003) used dynamic BBNs called Mass Detection tool to detect fraudulent claims, which then used a rule generator called Suspicion Building Tool.

The different types of fraud detection are: internal, insurance, credit card, and telecommunications fraud detection. Internal fraud detection consists in determining fraudulent financial reporting by management (Lin et al. 2003; Bell et al. 2000), and abnormal retail transactions by employees (Kim et al. 2003). There are four types of insurance fraud detection: home insurance (Bentley, 2000; Von Altrock, 1997), crop insurance (Little et al. 2002), automobile insurance fraud detection(Phua et al. 2004; Viaene et al. 2004; Brockett et al. 2002; Stefano et al. 2001; Belhadji et al. 2000), and health insurance (Yamanishi et al. 2004; Riedinger et al. 2002). A single meta-classifier(Phua et al. 2004) is used to select the best base classifiers, and then combined with these base classifiers' predictions to improve cost savings (stacking-bagging). Automobile insurance fraud detection data set was used to demonstrate the stacking-bagging problem. Credit card fraud detection refers to screening credit

applications (Wheeler et al. 2000), and/or logged credit card transactions (Foster et al. 2004; Fan, 2004; Chen et al. 2004; Chiu et al. 2004; Kim et al. 2002; Maes et al. 2002; Syeda et al. 2002). Telecommunications subscription data (Cortes et al. 2003; Cahill et al. 2002; Rosset et al. 1999; Moreau et al. 1997), and/or wired and wireless phone calls (Kim et al. 2003; Burge et al. 2001) are monitored. Credit transactional fraud detection has been presented by (Foster et al. 2004) and bad debts prediction (Ezawa et al. 1996). Employee/retail (Kim et al. 2003), national crop insurance (Little et al. 2002), and credit application (Wheeler et al. 2000). Literature focus on video-on-demand websites (Barse et al. 2003) and IP-based telecommunication services (McGibney et al. 2003). Online sellers (Bhargava et al. 2003) and online buyers (Sherman, 2002) can be monitored by automated systems. Fraud detection in government organisations such as tax (Bonchi et al. 1999) and customs (Shao et al. 2002) has also been reported.

We discuss below supervised data mining technique to detect crime using Bayesian Belief Networks, Decision trees, and Artificial Neural Networks.

2.1 Bayesian Belief Networks

Bayesian Belief Networks provide a graphic model of causal relationships on which class membership probabilities(Han et al. 2000) are predicted, so that a given instance is legal or fraud (Prodromidis, 1999). Naïve Bayesian classification assumes that the attributes of an instance are independent, given the target attribute (Feelders et al. 2003). The aim is to assign a new instance to the class that has the highest posterior probability. The algorithm is very effective and can give better predictive accuracy when compared to C4.5 decision trees and backpropagation (Domingos et al. 1996; Elkan et al. 2001). However, when the attributes are redundant, the predictive accuracy is reduced (Witten et al. 1999).

2.2 Decision Trees

Decision trees are machine learning techniques that express independent attributes and a dependent attribute in a tree-shaped structure that represents a set of decisions (Witten et al. 1999). Classification rules, extracted from decision trees, are IF-THEN expressions in which the preconditions are logically ANDed and all the tests have to succeed if each rule is to be generated. The related applications include the analysis of instances from drug smuggling, governmental financial transactions (Mena et al. 2003), and customs declaration fraud (Shao et al. 2002) to more serious crimes such as drug related homicides, serial sex crimes (SPSS, 2003), and homeland security(James et al. 2002; Mena et al. 2003). Data mining methods have solved security and criminal detection problems. [Mena, 2003] reviewed the subject (intelligent agents, link analysis, text mining, decision trees, self-organizing maps, machine learning, and neural networks) for security managers, law enforcement investigators, counter-intelligence agents, fraud specialists, and information security analysts. C4.5 (Quinlan et al. 1993) is used to divide data into segments based and to generate descriptive classification rules that can be used to classify a new instance.

C4.5 can help to make predictions and to extract crime patterns. It generates rules from trees (Witten et al.., 1999) and handles numeric attributes, missing values, pruning, and estimating error rates. C4.5 performs slightly better than CART and ID3 (Prodromidis, 1999) in terms of predictive accuracy. The learning and classification steps are generally fast(Han et al. 2000). However, performance decrease can occur when C4.5 is applied to large datasets. C5.0 shows marginal improvements to decision tree induction.

2.3 Artificial Neural Networks

Artificial Neural Networks represent complex mathematical equations with summations, exponentials, and parameters to copy neurons (Berry et al. 2000). They have been applied to classify crime instances such as burglary, sexual offences, and known criminals' facial characteristics (Mena et al. 2003b). Backpropagation neural networks can process a large number of instances with tolerance to noisy data and has the ability to classify patterns on which they have not been trained (Han et al. 2000). They are appropriate where the results of the model are more important (Berry et al. 2000). However, backpropagation require long training hours, extensive testing, retaining parameters like the number of hidden neurons, learning rate (Bigus, 1996).

3. APPLICATION

The steps in crime detection are: i) classifiers, ii) integrate multiple classifiers, iii) ANN approach to clustering, and iv) visualization techniques to describe the patterns.

3.1 Bayesian Network

Bayesian Network is a Directed Acyclic Graph, where each node represents a random variable and is associated with the conditional probability of the node given its parents. This model shows each variable in a given domain as a node in the graph and dependencies between these variables as arcs connecting the respective nodes. That is, all the edges in the graphical model are directed and there are no cycles.

For the purpose of fraud detection, we construct two Bayesian networks to describe the behavior of auto insurance. First, a Bayesian network is constructed to model behavior under the assumption that the driver is fraudulent (F) and another model under the assumption the driver is a legitimate user (NF), see Figure 3. The 'fraud net' is set up by using expert knowledge. The 'user net' is set up by using data from non fraudulent drivers. During operation user net is adapted to a specific user based on emerging data. By inserting evidence in these networks (the observed user behavior x derived from his toll tickets) and propagating it through the network, we can get the probability of the measurement x under two above mentioned hypotheses. This means, we obtain judgments to what degree an observed user behavior meets typical fraudulent or non-fraudulent behavior. These quantities we call p(x|NF) and p(x|F). By postulating the probability of fraud P(F) and P(NF) = 1-P(F) in general and by applying Bayes' rule, we get the probability of fraud, given

the measurement x,

$$P(F|x) = P(F)p(x|F)/\quad p(x)$$

where, the denominator p(x) can be calculated as

$$P(x) = P(F)p(x|F) + P(NF)p(x|NF)$$

The chain rule of probabilities is:

Suppose there are two classes C_1, C_2 for fraud and legal respectively. Given an instance
$X = (X_1, X_2, \ldots, X_n)$ and each row is represented by an attribute vector $A = (A_1, A_2, \ldots, A_n)$

The classification is to derive the maximum $P(C_i|X)$ which can be derived from Bayes' theorem as given in the following steps:

i) $P(\text{fraud}|X) = [P(\text{fraud} \mid X)\, P(\text{fraud})] / P(X)$

$P(\text{legal}|X) = [P(\text{legal} \mid X)\, P(\text{legal})] / P(X)$

As $P(X)$ is constant for all classes, only $[P(fraud \mid X)\, P(fraud)]$ and $[P(legal \mid X)\, P(legal)]$ need to be maximized.

ii) The class prior probabilities may be estimated by:
$P(\text{fraud}) = s_i / s$

Here, s is the total number of training examples and s_i is the number of training examples of class *fraud.*

iii) A simplified assumption of no dependence relation between attributes is made.

Thus,

$$P(X|\text{fraud}) = \prod_{k=1}^{n} P(x_k|\text{fraud})$$

and $$P(X|\text{legal}) = \prod_{k=1}^{n} P(x_k|\text{legal})$$

The probabilities $P(x_1|fraud)$, $P(x_2|fraud)$ can be estimated from the training samples:

$$P(x_k|\text{fraud}) = s_{ik} / s_i$$

Here, s_i is the number of training examples for class *fraud* and $s_{i\,k}$ is the number of training examples of class with value x_k for A_k

3.1.1 Application

We present Bayesian learning algorithm to predict occurrence of fraud. Using the "Output" classification results for Table 1, there are 17 tuples classified as legal, and 3 as fraud. To facilitate classification, we divide the age of driver attribute into ranges:

Table 1: Training set

Instance	Name	Gender	Age_ driver	fault	Driver_ rating	Vehicle_ age	Output
1	David Okere	M	25	1	0	2	legal
2	Beau Jackson	M	32	1	1	5	fraud
3	Jeremy Dejean	M	40	0	0	7	legal
4	Robert Howard	M	35	1	0.33	1	legal
5	Crystal Smith	F	22	1	0.66	8	legal
6	Chibuike Penson	M	36	0	0.66	6	legal
7	Collin Pyle	M	42	1	0.33	3	legal
8	Eric Penson	M	39	1	1	2	fraud
9	Kristina Green	F	29	1	0	4	legal
10	Jerry Smith	M	33	1	1	5	legal
11	Maggie Frazier	F	42	1	0.66	3	legal
12	Justin Howard	M	21	1	0	2	fraud
13	Michael Vasconi	M	37	0	0.33	4	legal
14	Bryan Thompson	M	32	1	0.33	4	legal
15	Chris Wilson	M	28	1	1	6	legal
16	Michael Pullen	M	42	1	0	5	legal
17	Aaron Dusek	M	48	1	0.33	8	legal
18	Bryan Sanders	M	49	1	0	3	legal
19	Derek Garrett	M	32	0	0	3	legal
20	Jasmine Jackson	F	27	0	1	2	legal
X	Crystal Smith	F	31	1	0	2	?

Table 2 shows the counts and subsequent probabilities associated with the attributes. With these simulated training data, we estimate the prior probabilities:

The classifier has to predict the class of instance to be fraud or legal.

$P(fraud) = s_i / s = 3/20 = 0.15$
$P(legal) = s_i / s = 17/20 = 0.85$

Table 2: Probabilities associated with attributes

Attribute	Value	Count		Probabilities	
		legal	fraud	legal	Fraud
Gender	M	13	3	13/17	3/3
	F	4	0	4/17	0/3
age_driver	(20, 25)	3	0	3/18	0
	(25, 30)	4	0	4/18	0
	(30, 35)	3	1	3/18	1/2
	(35, 40)	3	1	3/18	1/2
	(40, 45)	3	0	3/18	0
	(45, 50)	2	0	2/18	0
fault	0	5	0	5/17	0
	1	12	3	12/17	3/17
driver_rating	0	6	1	6/17	1/3
	0.33	5	0	5/17	0
	0.66	3	0	3/17	0
	1	3	2	3/17	2/3

We use these values to classify a new tuple. Suppose, we wish to classify X = (Crystal Smith, F, 31). By using these values and the associated probabilities of gender and driver age, we obtain the following estimates:

$P(X|legal) = 4/17 * 3/18 = 0.039$
$P(X|fraud) = 3/3 * 1/2 = 0.500$

Thus, likelihood of being legal = 0.039 *0.9=0.0351

Likelihood of being fraud = 0.500 *0.1= 0.050

We estimate $P(X)$ by summing up these individuals likelihood values since X will be either legal of fraud:

$P(X) = 0.0351 + 0.050 = 0.0851$

Finally, we obtain the actual probabilities of each event:

$$P(\text{legal} \mid X) = (0.039 * 0.9)/0.0851 = 0.412$$
$$P(\text{fraud} \mid X) = (0.500 * 0.1) / 0.0851 = 0.588$$

Therefore, based on these probabilities, we classify the new tuple as fraud because it has the highest probability.

Since attributes are treated as independent, the addition of redundant ones reduces its predictive power. To relax this conditional independence is to add derived attributes which are created from combinations of existing attributes.

Missing data cause problems during classification process. Naïve Bayesian classifier can handle missing values in training datasets. To demonstrate this, seven missing values appear in dataset.

The naïve Bayes approach is easy to use and only one scan of the training data is required. The approach can handle missing values by simply omitting that probability when calculating the likelihoods of membership in each class. Although the approach is straightforward, it does not always yield satisfactory results. The attributes usually are not independent. We could use subset of the attributes by ignoring any that are dependent on others. The technique does not handle continuous data. Dividing the continuous values into ranges could be used to solve this problem, but the division of the continuous values is a tedious task, and how this is done can impact the results.

3.2 DECISION TREE-BASED ALGORITHM

A decision tree (DT) is a tree associated with a database that has each internal node labeled with an attribute, each arc labeled with a predicate that can be applied to the attribute, and each leaf node labeled with a class. Solving the classification problem is a two-step process: i) decision tree induction-construct a DT, and ii) apply the DT to determine its class. Rules can be generated that are easy to interpret. They scale well for large databases because the tree size is independent of the database size.

DT algorithms do not easily handle continuous data. The attribute domains must be divided into categories. Handling missing is difficult. Since the DT is constructed from the training data, overfitting may occur. This can be overcome via tree pruning.

3.2.1 C4.5 Algorithm

The basic algorithm for decision tree is s follows:

i) Suppose there are two classes for fraud and legal. The tree starts as a single node *N* representing the training samples.

ii) If the samples are of the same class *fraud*, then the node becomes a leaf and is labeled as *fraud*.

iii) Otherwise, the algorithm uses an entropy-based measure as a heuristic for selecting the attribute that will best separate the samples into individual classes.

The entropy, or expected information needed to classify a given sample is:
I(fraud, legal)= - (NumberFraudSamples / NumberSamples)
$\log_2$ (NumberFraudSamples / NumberSamples) –
(NumberLegalSamples / NumberSamples)
$\log_2$ (NumberLegalSamples / NumberOfSamples)

iv) Expected information or entropy required to classify into subsets by test attribute E is:

E(A) = ∑ [(NumberTestAttributeFraudValues/ NumberSamples) + (NumberTestAttributeLegalValues/ NumberSamples)]* [I(TestAttributeFraudValues, TestAttributeLegalValues)]

v) Expected reduction in entropy is:

Gain(A)= I – E(A)

The algorithm computes the information gain of each attribute. The attribute with highest information gain is the one selected for test attribute.

vi) A branch is created for each known value of the test attribute. The algorithm uses the same process iteratively to form a decision tree at each partition. Once an attribute has occurred at a node, it need not be considered in any of the node's descendents.

The iterative partitioning stops only when one of the conditions is true: a) all examples for a given node belong to the same class, or b) there are no remaining attributes on which samples may be further partitioned. If this is the case, a leaf is created with the class in majority among samples, c) there are no samples for the branch test-attribute. In this case, a leaf is created with the majority class in samples

3.3 Rule Based Algorithm

One way to perform classification is to generate if-then rules. There are algorithms that generate rules from trees as well as algorithms that generate rules without first creating DTs.

3.3.1 Generating Rules from a Decision Tree

The following rules are generated for the Decision Tree (DT).

If driver age =25, then class = legal

If (driver_age =40) $\wedge$ (vehicle_age =7), then class = legal

If (driver_age =32)) $\wedge$ (driver_rating =1), then class = fraud

If (driver_age $\leq$ 40)) $\wedge$ (driver_rating =1)) $\wedge$ (vehicle_age =2), then class = fraud

If (driver_age > 40)) $\wedge$ (driver_age $\leq$ 50)) $\wedge$ (driver_rating = 0.33), then class = legal

4. MODEL PERFORMANCE

4.1 Confusion Matrix

There are two ways to examine the performance of classifiers: i) confusion matrix, and ii) to use a ROC graph. Given a class, C_j, and a tuple, t_i, that tuple may or may not be assigned to that class while its actual membership may or may not be in that class. With two classes, there are four possible outcomes with the classification as: i) true positives (hits), ii) false positives (false alarms), iii) true negatives (correct rejections), and iv) false negatives. False positive occurs if the actual outcome is legal but incorrectly predicted as fraud. False negative occurs when the actual outcome is fraud but incorrectly predicted as legal. A confusion matrix (Kohavi and Provost, 1998), Table 3a, contains information about actual and predicted classifications. Performance is evaluated using the data in the matrix. Table 3b shows confusion matrix built on simulated data. It shows the classification model being applied to the test data that consists of 7000 instances roughly split evenly between two classes. The model commits some errors and has an accuracy of 78%. We also applied the model to the same data, but to the negative class with respect to class skew in the data. The quality of a model highly depends on the choice of the test data. We also that that ROC curves are not so dependent on the choice of test data, at least with class skew.

Table 3a: Confusion Matrix

		Observed	
		legal	fraud
predicted	legal	TP	FP
	fraud	FN	TN

Table 3b: Confusion matrix of a model applied to test dataset

		Observed		
		legal	fraud	accuracy: 0.78
predicted	legal	3100	1125	recall: 0.86
	fraud	395	2380	precision: 0.70

A number of model performance metrics (Table 3c) can be derived from the confusion matrix.

Table 3c: Performance metrics

model performance metrics	
Accuracy(AC)	$AC = \frac{a+d}{a+b+c+d}$
Recall or true positive rate(TP)	$TP = \frac{d}{c+d}$
False positive rate(FP)	$FP = \frac{b}{a+b}$
True negative rate(TN)	$TN = \frac{a}{a+b}$
False negative rate(FN)	$FN = \frac{c}{c+d}$
Precision(P)	$P = \frac{d}{b+d}$
geometric mean(g-mean)	$g-mean_1 = \sqrt{TP*P}$ $g-mean_2 = \sqrt{TP*TN}$
F-measure	$F = \frac{(\beta^2+1)*P*TP}{\beta^2*P+TP}$

The *accuracy* determined in (Table 3b) may not be an adequate performance measure when the number of negative cases is much greater than the number of positive cases (Kubat et al.., 1998). Suppose there are 1500 cases, 1460 of which are negative cases and 40 of which are positive cases. If the system classifies them all as negative, the accuracy would be 97.3%, even though the classifier missed all positive cases. Other performance measures are *geometric mean* (*g-mean*) (Kubat et al.., 1998), and *F-Measure* (Lewis and Gale, 1994). For calculating F-measure, β has a value from 0 to ∞ and is used to control the weight assigned to *TP* and *P*. Any classifier evaluated using *g-mean* or *F-measure* will have a value of 0, if all positive cases are classified incorrectly.

To easily view and understand the output, visualization of the results is helpful.

Naïve Bayesian visualization provides an interactive view of the prediction results. The attributes can be sorted by the predictor and evidence items can be sorted by the number of items in its storage bin. Attribute column graphs help to find the significant attributes in neural networks. Decision tree visualization builds trees by splitting attributes from C4.5 classifiers.

Cumulative gains and lift charts are visual aids for measuring model performance. Lift is a measure of a predictive model calculated as the ratio between the results obtained with or without the predictive model. For instance, if 105 of all samples are actually fraud and a naïve Bayesian classifier could correctly predict 20 fraud samples per 100 samples, then that corresponds to a lift of 4.

Table 4: Costs of Predictions

fraud	legal
True Positive(Hit) cost= number of hits* average cost per investigation	False Positive(False alarm) cost=number of false alarms * (Average cost per investigation + average cost per claim)
False Negative(miss) cost= number of misses* average cost per claim	True Negative(correct rejection) cost = number of correct rejection claims * average cost per claim

Table 4 shows that True Positives (hits) and False Positives (false alarms) require cost per investigation. False alarms cost are the most expensive because both investigation and claim costs are required. False Negatives (misses) and True Negatives(correct rejection) are the cost of claim.

4.2 Relative Operating Characteristic Curve

Another way to examine the performance of classifiers is to use a Relative

Operating Characteristic (ROC) curve, (Swets, 1988). A ROC graph is a curve that depicts the performance and performance tradeoff of a classification model (Fawcett, 2004, Flach, 2004) with the False Positives along X -axis and the True Positives along the Y axis. The point (0, 1) is the perfect classifier: it classifies all positive cases and negative cases correctly. It is (0, 1) because the false positive FP is 0, and the TP rate is 1. The point (0, 0) represents a classifier that predicts all cases to be negative, while the point (1, 1) corresponds to a classifier that predicts every case to be positive. Point (1, 0) is the classifier that is incorrect for all classifications. An ROC curve or point is independent of class distribution or error costs (Provost et al.., 1998). It sums all information contained in the confusion matrix, since FN is the complement of TP and TN is the complement of FP (Swets, 1988). It provides a visual tool for examining the exchange between a classifier to correctly identify positive cases and the number of negative cases incorrectly classified.

We introduce to new performance metrics to construct ROC curves (in confusion matrix terms), the TP Rate (TPR) and the FP Rate (FPR):

TPR(recall) = TP / (TP+FN)

FPR = FP / (TN +FP)

The classifier is mapped to the same point in the ROC graph regardless of whether the original test set with sampled down negative class is used illustrating that ROC graphs are not sensitive to class skew.

One way of comparing ROC points is by using an equation that equates accuracy with the Euclidian distance from the perfect classifier, the point (0, 1). We include a weight factor that allows defining relative misclassification costs. We define AC_d as a distance based performance measure:

$AC_d = 1 - \sqrt{W * (1 - TP)^2 + (1 - W) * FP^2}$, where W ranges from 0 to 1, that is used to assign relative importance to false positives and false negatives. AC_d ranges from 0 for the perfect classifier to *sqrt(2)* for a classifier that classifies all cases incorrectly. AC_d differs from *g-mean*$_1$, *g-mean*$_2$ and *F-measure* in that it is equal to 0 only if all cases are classified correctly. In other words, a classifier evaluated using AC_d gets some credit for correct classification of negative cases, regardless of its accuracy in correctly identifying positive cases.

5. CONCLUSIONS

We studied the existing fraud detection systems. To predict and present fraud we used Naïve Bayesian classifier. We looked at model performance metrics derived from the confusion matrix. We illustrated how ROC curves can be deployed for model assessment. Performance metrics such as accuracy, recall,

and precision are derived from the confusion matrix. ROC analysis provides a highly visual account of a model's performance. It is strong with respect to class skew, making it a reliable performance metric in many important fraud detection application areas.

REFERENCES

Barse, E., Kvarnstrom, H. & Jonsson, E. (2003). Synthesizing Test Data for Fraud Detection Systems. *Proc. of the 19th Annual Computer Security Applications Conference*, 384-395.

Bell, T. & Carcello, J. (2000). A Decision Aid for Assessing the Likelihood of Fraudulent Financial Reporting. *Auditing: A Journal of Practice and Theory* **10**(1): 271-309.

Belhadji, E., Dionne, G. & Tarkhani, F. (2000). A Model for the Detection of Insurance Fraud. *The Geneva Papers on Risk and Insurance* **25**(4): 517-538.

Bentley, P. (2000). Evolutionary, my dear Watson: Investigating Committee-based Evolution of Fuzzy Rules for the Detection of Suspicious Insurance Claims. *Proc. of GECCO2000*.

Bentley, P., Kim, J., Jung., G. & Choi, J. (2000). Fuzzy Darwinian Detection of Credit Card Fraud. *Proc. of 14th Annual Fall Symposium of the Korean Information Processing Society.*

Bhargava, B., Zhong, Y., & Lu, Y. (2003). Fraud Formalization and Detection. *Proc. of DaWaK2003*, 330-339.

Bolton, R. & Hand, D. (2002). Statistical Fraud Detection: A Review (With Discussion). *Statistical Science* **17**(3): 235-255.

Bolton, R. & Hand, D. (2001). Unsupervised Profiling Methods for Fraud Detection. *Credit Scoring and Credit Control VII*.

Bonchi, F., Giannotti, F., Mainetto, G., Pedreschi, D. (1999). A Classification-based Methodology for Planning Auditing Strategies in Fraud Detection. *Proc. of SIGKDD99*, 175-184.

Brause, R., Langsdorf, T. & Hepp, M. (1999). Neural Data Mining for Credit Card Fraud Detection. *Proc. of 11th IEEE International Conference on Tools with Artificial Intelligence*.

Brockett, P., Derrig, R., Golden, L., Levine, A. & Alpert, M. (2002). Fraud Classification using Principal Component Analysis of RIDITs. *Journal of Risk and Insurance* **69**(3): 341-371.

Burge, P. & Shawe-Taylor, J. (2001). An Unsupervised Neural Network Approach to Profiling the Behavior of Mobile Phone Users for Use in Fraud Detection. *Journal of Parallel and Distributed Computing* **61**: 915-925.

Cahill, M., Chen, F., Lambert, D., Pinheiro, J. & Sun, D. (2002). Detecting Fraud in the Real World. *Handbook of Massive Datasets* 911-930.

Chan, P., Fan, W., Prodromidis, A. & Stolfo, S. (1999). Distributed Data Mining in Credit Card Fraud Detection. *IEEE Intelligent Systems* **14**: 67-74.

Chen, R., Chiu, M., Huang, Y. & Chen, L. (2004). Detecting Credit Card Fraud by Using Questionnaire-Responded Transaction Model Based on Support Vector Machines. *Proc. of IDEAL2004*, 800-806.

Cortes, C., Pregibon, D. & Volinsky, C. (2003). Computational Methods for Dynamic Graphs. *Journal of Computational and Graphical Statistics* **12**: 950-970.

Cox, E. (1995). A Fuzzy System for Detecting Anomalous Behaviors in Healthcare Provider Claims. In Goonatilake, S. & Treleaven, P. (eds.) *Intelligent Systems for Finance and Business*, 111-134. John Wiley.

Elkan, C. (2001). Magical Thinking in Data Mining: Lessons from CoIL Challenge 2000. *Proc. of SIGKDD01*, 426-431.

Ezawa, K. & Norton, S. (1996). Constructing Bayesian Networks to Predict Uncollectible Telecommunications Accounts. *IEEE Expert* October: 45-51.

Fan, W. (2004). Systematic Data Selection to Mine Concept- Drifting Data Streams. *Proc. of SIGKDD04*, 128-137.

Fawcett, T. (2004). ROC graphs: Notes and practical considerations for researchers. Machine Learning, 3.

Fawcett, T., & Flach, P. A. (2005). A response to web and Ting's on the application of ROC analysis to predict classification performance under varying class distributions. *Machine Learning,* 58(1): 33-38.

Flach, P. (2004). Tutorial at ICML 2004: The many faces of ROC analysis in machine learning. Unpublished manuscript.

Flach, P., Blockeel, H., Ferri, C., Hernandez-Orallo, J., & Struyf, J. (2003). Decision support for data mining: Introduction to ROC analysis and its applications. *Data mining and decision support: Aspects of integration and collaboration,* 81-90.

Flach, P. A. (2003). The geometry of ROC space: Understanding machine learning metrics through ROC isometrics. *Proceedings of the Twentieth International Conference on Machine Learning,* 194–201.

Foster, D. & Stine, R. (2004). Variable Selection in Data Mining: Building a Predictive Model for Bankruptcy. *Journal of American Statistical Association* **99**: 303-313.

He H, Wang J, Graco W and Hawkins S.(1997). Application of Neural Networks to Detection of Medical Fraud. Expert Systems with Applications, **13**, 329-336.

James F.(2002). FBI has eye on business databases. *Chicago Tribune*, Knight Ridder/ Tribune Business News.

Kim, H., Pang, S., Je, H., Kim, D. & Bang, S. (2003). Constructing Support Vector Machine Ensemble. *Pattern Recognition* **36**: 2757-2767.

Kim, J., Ong, A. & Overill, R. (2003). Design of an Artificial Immune System as a Novel Anomaly Detector for Combating Financial Fraud in Retail Sector. *Congress on Evolutionary Computation*.

Lin, J., Hwang, M. & Becker, J. (2003). A Fuzzy Neural Network for Assessing the Risk of Fraudulent Financial Reporting. *Managerial Auditing Journal* **18**(8): 657-665.

Little, B., Johnston, W., Lovell, A., Rejesus, R. & Steed, S. (2002). Collusion in the US Crop Insurance Program: Applied Data Mining. *Proc. of SIGKDD02*, 594-598.

Maes, S., Tuyls, K., Vanschoenwinkel, B. & Manderick, B. (2002). Credit Card Fraud Detection using Bayesian and Neural Networks. *Proc. of the 1st International NAISO Congress on Neuro Fuzzy Technologies*.

Magnify(2002). FraudFocus Advanced Fraud Detection, White Paper, Chicago.

Magnify(2002). The Evolution of insurance Fraud Detection: Lessons learnt from other industries, White Paper, Chicago.

Major, J. & Riedinger, D. (2002). EFD: A Hybrid Knowledge/Statistical-based system for the Detection of Fraud. *Journal of Risk and Insurance* **69**(3): 309-324.

Meena J(2003). Data mining for Homeland Security. Executive Briefing, VA.

Meena J(2003). Investigative Data Mining for Security and Criminal Detection, Butterworth Heinemann, MA.

McGibney, J. & Hearne, S. (2003). An Approach to Rules-based Fraud Management in Emerging Converged Networks. *Proc. Of IEI/IEEE ITSRS 2003*.

Moreau, Y. & Vandewalle, J. (1997). Detection of Mobile Phone Fraud Using Supervised Neural Networks: A First Prototype. *Proc. of 1997 International Conference on Artificial Neural Networks*, 1065-1070.

Ormerod T., Morley N., Ball L., Langley C., and Spenser C. (2003). 'Using Ethnography To Design a Mass Detection Tool (MDT) For The Early Discovery of Insurance Fraud', Computer Human Interaction*,* April 5-10, Ft. Lauderdale, Florida.

Phua, C., Alahakoon, D. & Lee, V. (2004). Minority Report in Fraud Detection: Classification of Skewed Data, *SIGKDD Explorations* **6**(1): 50-59.

Provost, F., Fawcett, T., & Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms. *Proceedings of the Fifteenth InternationalConference on Machine Learning,* , 445–453.

Rosset, S., Murad, U., Neumann, E., Idan, Y. & Pinkas, G. (1999). Discovery of Fraud Rules for Telecommunications - Challenges and Solutions. *Proc. of SIGKDD99*, 409-413.

SAS e-Intelligence(2000). Data Mining in the Insurance industry: Solving Business problems using *SAS Enterprise Miner Software*, White Paper.

Shao, H., Zhao, H. & Chang, G. (2002). Applying Data Mining to Detect Fraud Behavior in Customs Declaration. *Proc. of 1st International Conference on Machine Learning and Cybernetics*, 1241-1244.

Sherman, E. (2002). Fighting Web Fraud. *Newsweek,* June 10.

SPSS(2003). Data mining and Crime analysis in the Richmond Police Department, White Paper, Virginia.

Stefano, B. & Gisella, F. (2001). Insurance Fraud Evaluation: A Fuzzy Expert System. *Proc. of IEEE International Fuzzy Systems Conference*, 1491-1494.

Syeda, M., Zhang, Y. & Pan, Y. (2002). Parallel Granular Neural Networks for Fast Credit Card Fraud Detection. *Proc. of the 2002 IEEE International Conference on Fuzzy Systems*.

Swets, J. A., Dawes, R. M., & Monahan, J. (2000). Better decisions through science. *Scientific American; Scientific American, 283*(4), 82-87.

Von Altrock, C. (1997). Fuzzy Logic and Neurofuzzy Applications in Business and Finance. 286-294. Prentice Hall.

Viaene, S., Derrig, R. & Dedene, G. (2004). A Case Study of Applying Boosting Naive Bayes to Claim Fraud Diagnosis. *IEEE Transactions on Knowledge and Data Engineering* **16**(5): 612-620

Yamanishi, K., Takeuchi, J., Williams, G. & Milne, P. (2004). On-Line Unsupervised Outlier Detection Using Finite Mixtures with Discounting Learning Algorithms. *Data Mining and Knowledge Discovery* **8**: 275-300.

Weatherford, M.(2002). Mining for Fraud. *IEEE Intelligent Systems*, July/ August, 4-6.

Wheeler, R. & Aitken, S. (2000). Multiple Algorithms for Fraud Detection. *Knowledge-Based Systems* **13**(3): 93-99.

Williams, G. J. and Huang, Z.(1997). 'Mining the Knowledge: Mine the Hot Spots Methodology for Mining Large Real World Databases', 10th Australian Joint Conference on Artificial Intelligence, Published in Lecture Notes in Artificial Intelligence, Springer-Verlag, December, Perth, Western Australia.

Williams, G.(1999). 'Evolutionary Hot Spots Data Mining: An Architecture for Exploring for Interesting Discoveries', Proceedings of the 3rd Pacific-Asia Conference in Knowledge Discovery and Data Mining, Beijing, China.