International Journal of Economics, Commerce and Management
United Kingdom Vol. III, Issue 12, December 2015
http://ijecm.co.uk/ ISSN 2348 0386

A COMPARATIVE STUDY OF DECISION TREE ALGORITHMS FOR CLASS IMBALANCED LEARNING IN CREDIT CARD FRAUD DETECTION

Maira Anis ✉
PhD Scholar, School of Management and Economics
University of Electronic Science and Technology of China, China
maira7pk@hotmail.com

Mohsin Ali
PhD Scholar, School of Management and Economics,
University of Electronic Science and Technology of China, China
Employee, International Islamic University, Islamabad, Pakistan
mohsinali757@gmail.com, mohsinali@iiu.edu.pk

Amit Yadav
PhD Scholar, School of Management and Economics
University of Electronic Science and Technology of China, China
amitaryan2u@yahoo.com

Abstract

Credit card fraud detection along with its inherent property of class imbalance is one of the major challenges faced by the financial institutions. Many classifiers are used for the fraud detection of imbalanced data. Imbalanced data withhold the performance of classifiers by setting up the overall accuracy as a performance measure. This makes the decision to be biased towards the majority class that results in misclassifying the minority class. In today's revolutionary era of technology most transactions are based on the credit cards that make it more vulnerable to fraud. Credit card data is naturally an imbalanced data and it has been found that most of the classifiers perform poorly on the credit card imbalanced data. Resampling is a technique that deals with the imbalanced data. The aim of this paper is to find the best *distribution among the classifiers, to get insights of credit card data by random under sampling (RUS) along with feature selection and conclude about a useful model that can measure the credit card fraud risk more efficiently. We applied RUS with feature selection for the family of Decision Tree classifier. Results showed that the given models improved the performance for the Decision Tree classifiers used in a previous study.*

Keywords: Imbalanced Data Set, Resampling, Classification, Performance evaluation, Credit Card Fraud Detection

INTRODUCTION

E-commerce is becoming an essential tool for global trade in today's electronic world. The use of credit card has been boomed to its peak due to the recent advancements in e-commerce. This rapid advancement has created an attractive source of revenue for criminals and has unfortunately lead them to fraud. Uses of credit card fraud have been increased dramatically in the past two decades. Globally, the total number of credit cards circulating in 2011 was 2,039.3 m, and these were used in 64.24 billion transactions (Neilson 2012). There have been impressive growths in figures of the number of credit cards in the recent past (Akkoc 2012). This is because of existing security weaknesses in traditional credit card processing systems that result in loss of billions of dollars every year. Fraudsters are now using sophisticated techniques to perpetrate credit card fraud. These fraudulent activities present unique and global challenges to banks and other financial institutions who issue credit cards. In case of bank cards, a study done by American Bankers Association in 1996 reveals that the estimated gross fraud loss was $790 million in 1995 (Roberds 1998). The majority of the loss due to credit card fraud is suffered by the USA alone. This is not surprising since 71% of all credit cards are issued only in the USA. In 2005, the total fraud loss in the USA was reported to be $2.7 billion and it has gone up to $3.2 billion in 2007 (Statistics for General and Online Card Fraud 2007). Another survey in 2007 of over 160 companies revealed that online fraud (committed over the Web or phone shopping) is 12 times higher than offline fraud (committed by using a stolen physical card) ("sell it on the web" Online fraud is 12 times higher than offline fraud 2007).

Imbalanced data is found in many real world problems. A data is called imbalanced when number of instances from one class is being outnumbered by the instances of the other class (Japkowicz 2000).The detection of credit card fraud is a complex computational task. Many Classifiers have been used by the machine learning community to reduce the amount of fraud. As the credit card data is inherited an imbalanced data so it naturally makes the classifiers to be overwhelmed by the majority class i.e. non-fraudulent class. There is no single classifier that works well with credit card fraud detection as they all only predict the possibility of the transaction to be fraudulent. While there are some key points that should be considered in choosing the classifier. The characteristics in choosing the classifier for the credit card fraud detection system are;

i. The classifier should identify the frauds accurately i.e. the true positive rate should be high.
ii. The classifier should detect the frauds quickly.
iii. The classifier should not predict a genuine transaction as fraud i.e. the false positive rate should be low.

There are many techniques to handle with imbalanced data. Clever resampling and combination methods can give surprising results in unleashing the hidden truths behind the imbalanced data (Chawla et al 2002 & 2003, Batista et al 2004, Gou & Viktor 2004 and Kubat & Matwin 1997). In this paper our aim is to find the right distribution and to get the best Decision Tree classifier that work efficiently with the feature selection and random under sampling (RUS) for the credit card data.

A lot of research has been done to cater the problems caused by the imbalanced credit card data. Section 2 describes some benchmarking studies in the field of credit card fraud detection. Section 3 fives an explanatory over view of the classification techniques deployed for this study. Section 4 will describe the datasets and the experimental design of the study. Section 5 briefly discusses and explains the results while section 6 concludes them.

RELATED WORK

An extensive study have been done in the field of fraud detection ranging from supervised learning and unsupervised learning to hybrid models. There is no specific compiled study about the decision tree classifiers. This study will focus on the benchmarking studies that applied classification techniques in the domain of credit card fraud detection.

Credit card fraud detection has drawn a lot of research interest and a number of techniques, with special emphasis on data mining and neural networks. Ghosh & Reilly (1994) proposed a framework for credit card fraud detection with a neural network. They built a detection system, which was trained on a large sample of labeled credit card account transactions. They used a neural network system which consists of a three-layered feed-forward network with only two training passes to achieve a reduction of 20% to 40% in total credit card fraud loses. This system also significantly reduced the investigation workload of the fraud analysts. Chan & Stolfo (1997) suggested a credit card fraud detection system using Meta learning techniques to learn models of fraudulent credit card transactions. Meta learning is a general strategy that provides a means for combining and integrating a number of separately built classifiers or models. Dorronsoro et al (1997) developed a neural network based fraud detection system called Minerva. This system's main focus is to imbed itself deep in credit card transaction servers to detect fraud in real-time. Kokkinaki (1997) created a user profile for each credit card account and tested incoming transactions against the corresponding user's profile. He proposed a Similarity Tree algorithm, a variation of Decision Trees, to capture a user's habits.

Chan and Stolfo (1999) studied the class distribution of a training set and its effects on the performance of multi-classifiers on the credit card fraud domain. They used an agent based approach with distributed learning for detecting frauds in credit card transactions. It is based on artificial intelligence and combines inductive learning algorithms and meta learning methods for achieving higher accuracy.

Ehramikar (2000) showed that the most predictive Boosted Decision Tree classifier is one that is trained on a 50:50 class distribution of fraudulent and legitimate credit card transactions. Brause et al. (1999 a & b) have developed an approach that involves advanced data mining techniques and neural network algorithms to obtain high fraud coverage.

Syeda et al (2002) have used parallel granular neural networks (PGNNs) to improve the speed of data mining and knowledge discovery process in credit card fraud detection. Kim and Kim (2002) have identified skewed distribution of data and mix of legitimate and fraudulent transactions as the two main reasons for the complexity of credit card fraud detection. Chiu and Tsai (2004) identified the problem of credit card transaction data having a natural skew-ness towards legitimate transactions. Fan (2004) proposed an efficient algorithm based on decision trees. The decision tree "sifts through" old data and combines it with new data to construct the optimal model. Foster and Stine (2004) tried to predict personal bankruptcy using a fully automated stepwise regression model.in the banking industry. Phua et al (2004) suggest the use of meta classifier similar to Chan & Stolfo (1997) in fraud detection problems. They considered Naive Bayesian, C4.5, and Back Propagation neural networks as the base classifiers. Vatsa et al (2005) have proposed a game-theoretic approach to credit card fraud detection. They model the interaction between the attacker and Fraud Detection System (FDS) as a multistage game between two players, in which each player tries to maximize his payoff.

Chen et al (2004) presented a new method to address the credit card fraud problem. A questionnaire-responded transaction data of users was developed by using an online questionnaire. The support vector machine algorithm was then applied to decide if new transactions were fraudulent or legitimate. Abdelhalim & Traore (2009) tackled the application fraud problem where a fraudster applies for an identity certificate using someone else's identity. Sahin & Duman (2011) developed classification models, Decision Trees and SVM for the credit card fraud. This was the first study to compare the results for the decision trees and SVM. Seeja & Masoumeh (2014) gave a novel method of credit card fraud detection based on frequent item set mining.

Decision Tree and its Family of Classifiers

Decision Tree is a non-parametric supervised approach used for classification and regression. The goal is to create a predictive model for the target values that can learn from the simple decision rules that are inferred from the attributes. Decision tree represent a simple tree like structure where non-terminal nodes represent test on attributes and terminal nodes represent the decision outcomes. An ordinary tree consists of root, nodes, branches and leaves.

The decision trees studied for this classification analysis will be implemented in WEKA (The Waikato Environment for Knowledge Analysis). WEKA is a tool used for data analysis of different machine learning algorithms. It includes implementation of data pre-processing, classification, clustering, association rules, regression and visualization of different algorithms. An overview of the classifiers used for this study is summarized below.

Decision Trees-Stump

Decision stump is basically a one level decision tree (Wayne & Langley 1992). In this algorithm the split is done at the root level that is based on a specific attribute or value of a pair. Stump is a decision tree with one root (internal) node that is connected to the terminal nodes. The algorithm makes predictions based on the value of the single attribute. Decision stump is also called 1-rules (Holte & Robert 1993). The term "decision stump" was devised in a 1992 ICML paper by Wayne & Langley (1992) and Oliver & Hand (1994).

Decision Trees-Random Forest

Random forest algorithm was developed by Breiman (2001). It is an ensemble of unpruned classification or regression trees. It combines Breiman’s method of bagging with random feature selection. These trees are induced from bootstrap samples of the training data with random feature selection in the tree induction process. Prediction in this tree is made by aggregating (majority vote for classification or averaging for regression) the predictions of the ensemble. Random forest generally exhibits a substantial performance improvement over the single tree classifier such as CART and C4.5. It has a generalization error rate that is favorably compared to Adaboost even though being more robust to noise.

a. J48

J48 is implementation of C4.5 tree in Weka. For C4.5 it creates pruned or unpruned tree. C4.5 is an algorithm that is used to create a decision tree developed by Quinlan (1986). C4.5 is an extension of ID3 algorithm. The decision trees generated by C4.5 are used for classification. C4.5 is also stated as a statistical classifier. The decision tree generated by this algorithm is done by the recursive partitioning of data. This algorithm uses a depth-first strategy to build trees and considers all possible tests that split the data set. Then the selection of test set is based on the set that gives the best information gain. For each discrete attribute, one test with outcomes with as many as the number of distinct values of the attribute is considered. For each continuous attribute, binary sets involving every distinct value of the attribute are considered. In order to gather the entropy gain of all these binary tests efficiently, the training dataset belonging to the node in consideration is sorted for the values of the continuous attribute and the entropy gains of the binary cut based on each distinct values are calculated in one scan of the sorted data. The process is repeated for each continuous attribute. Detailed introduction can be found in Quinlan (1986).

b. REP Tree

REP Tree builds a fast decision tree for classification and regression. It uses information gain as the splitting criteria, prunes it using reduced error pruning. It sorts numeric value attributes once while for the missing attributes it adopts the method of c4.5 method by Quinlan (1986).

c. Random Tree

Random Tree is a tree created at random from a set of possible trees with k random features at each node. The term “at random” implies that each tree in the set of trees has an equal chance of being sampled. It also means that the distribution of trees is “uniform”. Random trees are generated efficiently and with combination of large sets of random trees, it generally leads to accurate models. Random tree models have been comprehensively developed in the field of Machine Learning in the recent years.

d. Logistic Model Trees

Logistic Model Tree algorithm or LMT for short is combination of the logistic regression model with tree induction, and thus is an analogue of model trees for classification problems. LMT retains the standard decision tree structure with the logistic regression function at its leaf nodes. For a nominal attribute with k values, the node has k child nodes, and instances are sorted down one of the k branches depending on their value of the attribute. For numeric attributes, the node has two child nodes and the test consists of comparing the attribute value to a threshold: an instance is sorted down the left branch if its value for that attribute is smaller than the threshold and sorted down the right branch otherwise. For detailed theoretical information Landwehr et al., (2005) is referred.

RESEARCH METHODOLOGY

Experimental design and datasets

Data collections for credit cards are major problem for the researchers. For this reason, two data sets of credit card have been selected which are physically available (UCI repository), to analyze credit card fraud detection. The aim of this paper is to find the best classifier that work well under different distribution of credit card datasets with random under sampling and feature selection. The characteristics of datasets are summarized in Table 1.

Table 1: Characteristics of the credit card Datasets

Dataset	Features	DataSize	TrainingSize	TestingSize	Imb ratio
Aus	14	547	366	181	70/30
Gem	21	700	668	332	70/30

Resampling and performance evaluation metric

With very limited number of credit card datasets the task is to understand the imbalance foundations. For this, it is important to reduce the number of bad observations for both the datasets. This was done by the under sampling of the good observations.

Experiments conducted to compare various classification algorithms are implemented within the WEKA framework (Witten and Frank, 2005). Implementation of Decision Stump, Random Forest, J48, Random Tree, REP Tree and LMT is implemented in WEKA with default parameters. The data will be iteratively partitioned to different imbalanced ratios like 99/1,97.5/2.5, 95/5, 90/10, 80/20, 70/30, 60/40 and 50/50. Data will be divided in to training and testing sets. The training set will also use the cross validation scheme. In this experiment, the accuracy on the random sample has been obtained using 10-fold cross validation, which is helpful to prevent overfitting. In the following, accuracy is an average of any 9/10 sample as training set and the rest as testing set for 10 times.

For this study the number of bad observations in each training data was reduced artificially by a factor of 50% to 1%. By this reduction we got 9 datasets for each of the original dataset with different imbalanced ratios. The percentage splits used for this study are 50%, 60%, 70%, 80%, 85%, 90%, 95%, 97.5%, and 99%. The performance of the decision tree classifiers is evaluated by the Receiver Operating Curve (ROC) statistic that was proposed by Baesens et al(2003). ROC is a 2D- plot of TP (y-axis) against FP (x-axis) for the performance of classifiers. ROC is a graphical tool to distinguish the suboptimal classifiers from the optimal classifiers.

Another performance evaluation measure used is F-measure also called F-score or F1 metric. It is a weighted harmonic mean of Precision (P) and recall(R). It was used in information retrieval(1999) and is defined as;

$$F - measure = \frac{2(R * P)}{R + P}$$

Recall is the ratio of correctly classified positive (minority class) instances to total positive and precision is the ratio of correctly classified positive instances to total predicted as positive.

Feature selection and classification

Feature selection is a preprocessing step that is implemented to eliminate the irrelevant features from the dataset. Grall-Maes (2002) conducted a study on credit card transaction and found a high fraud catching rate with removal of highly correlated features. Feature elimination is performed for many reasons e.g. to reduce the training time of the algorithm, to make the models simpler to be visualized by the researcher and to generalize the data to reduce over fitting. For the datasets of this study feature selection is performed in WEKA interface. For the German credit card dataset, 4 attributes were left out of 21 attributes. While for Australian credit approval data set 8 attributes were left out of 15.

After feature selection the data is partitioned in to training and testing datasets with different imbalanced ratios. The training set is further randomly under sampled for the majority class to get the balanced distribution for the two classes. Training set is then used to build a model classifier for nine different imbalanced ratios using stratified 10-fold cross-validation. Stratified cross validation is performed for the reason to ensure that each fold has the right proportion of each class value. With this method we build a classifier in which each data point is used k-1 times for training and at least one time for testing. Cross validation is the way of reducing the variance in the data.

During the training process, for every decision tree classifier (except the Decision Stump, for decision stump results did not change for any value of seed), the model for the tree was selected for which cross validation random seed $[1,2,3,4,5,6,7,8,9]$ gave the most correctly classified instances. With different values of seeds there was improvement of 0.1 to 4 percent in correctly classified instances. The reason for selecting the model with most correctly classified instances is to learn a classifier with increased TP rate that resulted in high ROC area of the model (in the hold out set), that is the building block in the model selection. The results of this paper can be compared for German and Australian credit datasets too Brown & Mues (2012). Author used C4.5 and Random Forest (with other classifiers) for different distributions of the data Brown & Mues (2012). For Random Forest and C4.5 we got more improved results.

For the classifier C4.5 pruned trees were generated and the confidence level was varied from 0.01 to 0.25 to get the most suitable value during the 10-fold cross validation i.e. with high ROC area. The classifier Decision Stump required no parameter setting as with varied values of parameters, the results turned out to be same. For LMT, REPTree, RandomTree the seed value was varied during the validation process. While for the Random Forest, parameters tuned were the number of tress and the number of attributes along with the seed value described above. Range of trees examined during the validation process was [10,20,30,50,100,200,300,500] and the number of attributes selected per tree was [0.5,1,2]. Random Forest was also built using the WEKA package.

Statistical comparison of classifiers

A wide range of tests is presented to researchers for statistical comparison of classifiers. Among them some are non-parametric while others are parametric. Overall non-parametric tests are considered to be more suitable in comparing the classifiers as they do not consider the normal distribution and assume limited commensurability. For this study Wilcoxon Rank Sum Test will be used for the comparison Roc measures taken from six classifiers for two credit data sets. Wilcoxon Test (Wilcoxon & Wilcox, 1964) is a non-parametric alternative to a two sample t-test. It is also known as Wilcoxon Mann Whitney U test or simply U test. Wilcoxon Test takes in to account two independent samples to rank the observations from both samples simultaneously and assign average ranks in case of ties. The W statistic is calculated using the rank sum of smallest sample and is calculated as follows;

$$W = \sum_{i=1}^{n} R_i$$

In addition to ROC measure for the distributions, table 1 to 9 represents the ranks and the rejection (h=1) or acceptance (h=0) of null hypothesis along with the p-value. Table 1 to Table 9 shows the ranked performances of the classifiers under each distribution with its F statistic to understand clearly about the techniques that mark a significant difference to the best performing classifiers. Statistics for the Decision Tree classifiers, %CCL, F-measure and ROC area for the test sets as mentioned in figure 1 & 2.

EMPIRICAL RESULTS & DISCUSSION

Table 2 to Table 10 represents the percentage of correctly classified instances (%CCL), ROC area or AUC and F-measure for all the six classifiers along with different degrees of imbalance for Australian Credit data and German Credit data sets. The classifier with the highest AUC in each of the distribution is marked bold for each dataset. Analyzing keenly about the statistics we got from the classifiers, it is revealed that the two datasets performed differently for the same classifiers even having the same distributions. Firstly the results we got for the Australian data sets were found to be with more correctly classifies instance, high ROC area and F-measure. Among the six classifiers with increase in imbalance ratio the classifiers tend to behave almost equally. With increased imbalance ratio the classifiers LMT and RF gave equivalent results for the Australian data while for the German Credit data the results for J48 and RF were comparable. The results can be easily visualized from the ROC curves for both the datasets.

Table 2: ROC measure for the distribution 50:50

50% bad observations	Australian Data				German Data			
h=1, p=0.0043, W=45	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	87.8	88.3	82.8	8.5	87.8	70.1	53.1	5
LMT	87.86	88.3	88.0	8.5	68.0	74.0	53.5	6
REPTree	87.81	88.3	88.0	8.5	67.0	69.0	53.5	3.5
Stump	87.9	88.3	81.9	8.5	60	67.0	53.0	2
RandomTree	96.5	97.8	96.7	11	69.0	66.7	49.8	1
RandomForest	96.5	99.1	96.7	12	67.0	69.0	50.7	3.5

Table 3: ROC measure for the distribution 50:40

40% bad observations	Australian Data				German Data			
h=1, p=0.0022, W=57	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	85.8	90.6	87.2	10	68.5	68.8	76.8	5
LMT	84.7	93.1	84.6	12	65.25	73.3	73.2	6
REPTree	87.81	88.3	88.0	9	61.0	68.1	68.0	4
Stump	83.3	84.0	82.4	8	58.5	65.4	64.1	2
RandomTree	78.2	77.7	79.0	7	67.25	64.4	75.3	1
RandomForest	84.4	90.9	84.5	11	67.75	66.5	76.5	3

Table 4: ROC measure for the distribution 70:30

30% bad observations	Australian Data				German Data			
h=1, p=0.0022, W=57	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	92.2	93.5	92.7	11	65.3	64.4	46.4	3
LMT	85.5	94.3	85.1	12	64.3	70.5	50.2	6
REPTree	85.5	86.1	81.1	8	58.3	65.5	48.6	5
Stump	87.6	88.1	86.2	9	57.6	64.0	50.6	1
RandomTree	82.12	81.8	82.6	7	69.0	64.3	50.3	2
RandomForest	86.4	92.6	86.8	10	67.0	65.2	49.2	4

Table 5: ROC measure for the distribution 80:20

20% bad observations	Australian Data				German Data			
h=1, p=0.0022, W=57	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	86.2	88.5	88.9	9	62	61.1	47.2	1.5
LMT	86.9	95.9	86.4	12	64	68.9	50.0	6
REPTree	88.4	94.5	88.2	11	62	67.6	47.2	5
Stump	75.1	79.3	77.3	8	57	64.1	50.6	3
RandomTree	72.4	72.7	71.6	7	65	61.1	47.0	1.5
RandomForest	84.0	91.8	83.1	10	62	67.0	47.2	4

Table 6: ROC measure for the distribution 85:15

15% bad observations	Australian Data				German Data			
h=1, p=0.0022, W=57	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	96.1	97.1	96.2	12	63.3	68.6	47.6	5
LMT	87.5	95.5	86.3	11	62.6	70.7	50.0	6
REPTree	87.5	87.5	86.0	8.5	57.3	65.3	51.5	4
DECStump	87.5	87.5	86.0	8.5	57	64.1	50.6	2
RNDTree	79.8	79.8	80.8	7	65	61.1	47.0	1
RNDForest	87.5	92.8	86.6	10	62.6	64.6	47.2	3

Table 7: ROC measure for the distribution 90:10

10% bad observations	Australian Data				German Data			
h=1, p=0.0022, W=57	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	85.6	92.4	86.7	10	67	71.1	60.0	4
LMT	91.3	96.3	90.9	12	63	71.9	51.9	5
REPTree	89.8	90.0	88.9	8.5	67	73.7	54.8	6
Stump	89.8	90.0	88.9	8.5	61	69.3	58.1	3
RandomTree	84.0	84.1	83.6	7	70	66.9	55.9	1
RandomForest	88.4	94.4	87.9	11	69	68.5	58.7	2

Table 8: ROC measure for the distribution 95:5

5% bad observations	Australian Data				German Data			
h=0, p=0.0649, W=51	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	94.2	**95.4**	85.8	11	76	**91.7**	62.5	8
LMT	88.6	93.4	86.7	10	78	88.9	66.7	7
REPTree	88.5	93.1	86.7	9	68	85.2	57.9	5
Stump	88.6	87.5	75.0	6	64	76.3	57.1	1
RandomTree	82.8	82.7	81.3	3	76	80.7	64.7	2
RandomForest	94.3	**98.5**	87.5	12	74	**84.8**	62.9	4

Table 9: ROC measure for the distribution 97.5:2.5

2.5% bad observations	Australian Data				German Data			
h=0, p=0.1775, W=48	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	88.9	**90.0**	80.0	4	80	**86.0**	66.7	2
LMT	88.8	100.0	88.9	12	80	94.5	66.7	8
REPTree	88.8	97.5	88.9	10	80	96.5	66.7	9
DECStump	88.8	90.0	88.9	4	64	77.5	52.6	1
RNDTree	94.4	94.4	94.4	7	72	90.0	58.8	4
RNDForest	88.8	**99.4**	88.9	11	76	**92.5**	62.5	6

Table 10: ROC measure for the distribution 99:1

1% bad observations	Australian Data				German Data			
h=0, p=0.5130, W=43.5	% CCL	ROC area	F-measure	Rank	% CCL	ROC area	F-measure	Rank
J48	85.7	**89.3**	75.0	7	80	**87.5**	66.7	4.5
LMT	85.7	99.0	85.7	11	80	98.9	66.7	10
REPTree	85.7	87.5	85.7	4.5	60	78.1	50.0	2
Stump	85.7	87.5	85.7	4.5	60	75.0	50.0	1
RandomTree	85.7	87.5	85.7	4.5	90	92.7	49.9	8
RandomForest	85.7	**99.1**	85.7	12	80	**93.8**	50.0	9

Starting with the 50% distribution the results of AUC revealed the significance of Random Forest and LMT, these two classifiers ruled out the other classifiers for both the data sets. J48 was the second best in both datasets for 50% distribution. On 40% bad observations Random Tree was the worst classifier. Results for Random Tree and Decision Stump were almost same. At the original split of 70/30 J48, LMT and Random Forest performed significantly well than the other ones with highest AUC areas of 93.5, 70.5 and 92.6.

Overall, Decision stump, Random Tree and REPTree were the worst classifiers in line. On 15 percent bad observations Decision Stump and REPTree performed equivalently. With a split of 90/10, LMT was ahead with ROC score of 96.3 and 71.9 for both the data sets. After that Random Forest and J48 were in a row.

In a nutshell, it has been noticed that with great imbalance in datasets Random Forest and LMT were the leading classifier and performed well in terms of ROC and percentage of correctly classified instances at each distribution. Along with Random Forest and LMT, J48 gave the promising results to be a best classifier among the family of classifiers. It has also been noticed that with increase in the degree of skewness there was not enough evidence to reject the null hypothesis i.e. there was no significant difference between the ROC areas of the distribution examined at the level of each classifier.

The results in this study for the classifiers Random Forest and J48 have showed improved ROC area that was presented in Brown & Mues (2012). The results we found in this study promote Random Forest and LMT as the best classifiers in almost all the distributions.

Figure 1: ROC Curve for Australian Credit data sets

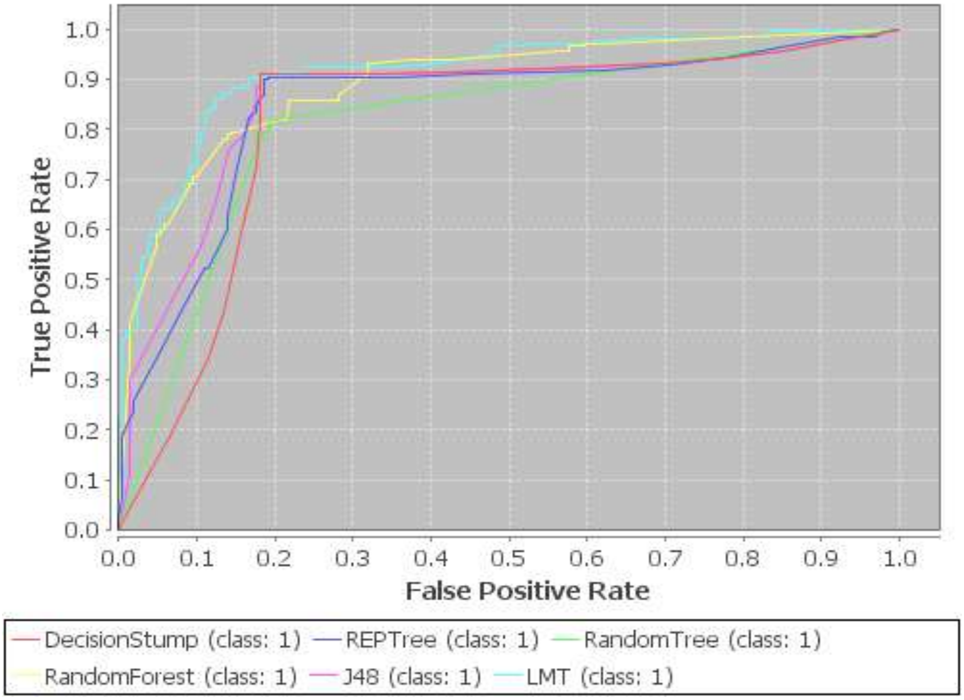
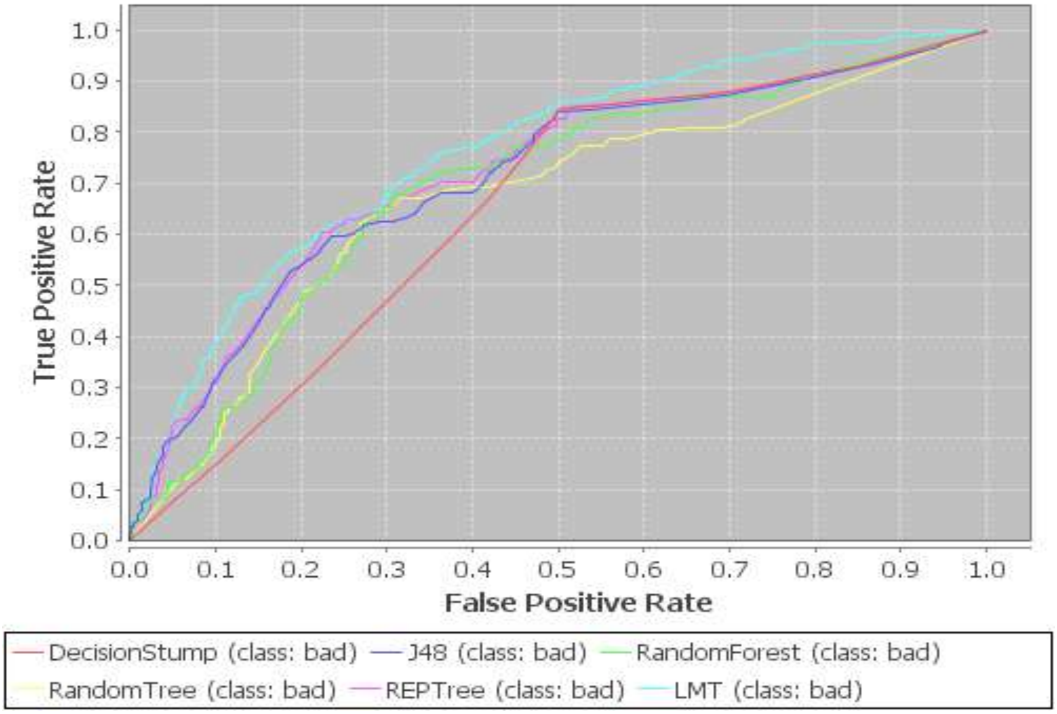


Figure 2: ROC Curve for German Credit data Set



CONCLUSION

With this comparative study of six Decision Trees classifier's over two datasets Random Forest is best classifiers among the other that we have used in this study. LMT mostly out rule the J48 classifier, comparison for the assessment was based on the AUC, F-measure and percentage of correctly classified. Further we applied the Wilcoxon Test to check the differences between the ROC areas of both samples.

Results show that when the imbalance ratio increases gradually in the data, Random Forest and LMT try to perform very well. With high imbalance ratio LMT perform better than J48 in terms of ROC area and F-measure. Also in the ranking procedure of the Wilcoxon Test the classifier assigned with highest ranks were Random Forest, LMT and J48.We can conclude that the choice of stump cannot be a good option for the data sets where we have great degree of skewness. As the LMT and Random Forest gave better results so there is need to explore them more with larger datasets using these findings, further extension to this work can be to apply different resampling techniques on the data to find more insights for the credit card imbalanced data and get more improved results.

REFERENCES

Abdelhalim, A. & Traore, I. (2009). "Identity Application Fraud Detection using Web". International Journal of Computer and Network Security 1, no. 1: 31-44.

Akkoc, S. (2012). An empirical comparison of conventional techniques, neural networks and the three stage hybrid Adaptive Neuro Fuzzy Inference System (ANFIS) model for credit scoring analysis: The case of Turkish credit card data. European Journal of Operational Research, 222(1), 168-178.

Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). Benchmarking state-of-the-art classification algorithms for credit scoring. Journal of the Operational Research Society, 54(6), 627–635.

Batista. G. E., Prati, R. C. & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. SIGKDD Explorations, 6(1):20-29.

Brause, R., Langsdorf, T. & Hepp, M. (1999a). Credit card fraud detection by adaptive neural data mining, Internal Report 7/99 (J. W. Goethe-University, Computer Science Department, Frankfurt, Germany).

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

Breiman, L. (2001). Random forests, Machine Learning 45(1), 5-32.

Brown. I., Mues. C, (2012). An experimental comparison of classification algorithms for imbalanced credit scoring data sets. Expert Systems with Applications 39, 3446–3453.

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

Chan, P. & Stolfo, S. (1997). On the Accuracy of Meta learning for Scalable Data Mining, J. Intelligent Information Systems, 8:5-28.

Chawla, N. V, Hall.L.O, Bowyer.K.W and Kegelmeyr.W.P. SMOTE: Synthetic Minority oversampling Technique. Journal of Artificial Intelligence Research, 16:321-357, 2002

Chawla, N. V., Lazarevic, A., Hall, I. O. & Bowyer, K. (2003). Smote Boost: Improving Prediction of the minority class in boosting. In Proceedings of Seventh European Conference on Principles and Practice of Knowledge Discovery in Databases, Pages 107-119, Dubrovnik, Croatia.

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 IDEAL 2004, 800-806.

Chiu, C. & Tsai, C. 2004. A Web Services-Based Collaborative Scheme for Credit Card Fraud Detection. Proc. of 2004 IEEE International Conference on e-Technology, e-Commerce and e- Service.

Dorronsoro, J., Ginel, F., Sanchez, C. & Cruz, C. (1997). Neural Fraud Detection in Credit Card Operations. IEEE Transactions on Neural Networks 8(4): 827-834.

Ehramikar, S. (2000). "The Enhancement of Credit Card Fraud Detection Systems using Machine Learning Methodology." MASc Thesis, Department of Chemical Engineering, University of Toronto.

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

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

Ghosh & Reilly, D. L. (1994). “Credit Card Fraud Detection with a Neural-Network,” Proc. 27th Hawaii International Conference on System Sciences: Information Systems: Decision Support and Knowledge-Based Systems, vol. 3, pp. 621-630.

Gou, H. & Viktor, H. I. (2004). Learning from imbalanced data sets with boosting and data generation: The Data Boost-IM approach. SIGKDD Explorations, 6(1):30-39.

Grall-Maes, E., Beauseory, P. (2002). Mutual information-based feature extraction on the time frequency plane. IEEE Transactions on Signal Processing 50(4) 779-790.

Holte, & Robert, C. (1993). "Very Simple Classification Rules Perform Well on Most Commonly Used Datasets".

Japkowicz, N. (2000). Learning from imbalanced data sets: A comparison of various strategies. In AAAI workshop on learning from imbalanced data sets (Vol. 6, pp.10–15).

Kim, M. & Kim, T. 2002. A Neural Classifier with Fraud Density Map for Effective Credit Card Fraud Detection, Proc. Of IDEAL 2002, 378-383.

Kokkinaki, A. (1997). On Atypical Database Transactions: Identification of Probable Frauds using Machine Learning for User Profiling. Proc. of IEEE Knowledge and Data Engineering Exchange Workshop, 107-113.

Kubat, M. & Matwin, S. (1997). Addressing the curse of imbalanced training sets: One sided selection. In Proceedings of the Fourteenth International Conference on Machine Learning, Pages 179-186, Nashville, Tennesse. Morgan Kaufmann.

Landwehr, N., Hall, M. & Frank, E. (2005). Logistic Model Trees. Machine Learning 59: 161.

Neilson (2012). Global Cards — 2011. The Nielsen Report, April 2012 (Issue 992), Carpinteria, CA, USA.

Oliver, J. & Hand, D. (1994); Averaging Over Decision Stumps, in Machine Learning: ECML-94, European Conference on Machine Learning, Catania, Italy, April 6–8, 1994, Proceedings, Lecture Notes in Computer Science (LNCS) 784, Springer, pp. 231–241

Online fraud is 12 times higher than offline fraud, 20 June, 2007. <http://sellitontheweb.com/ezine/news0434.shtml>.

Phua, C., Alahakoon, D., & V Lee. 2004. ‘Minority Report in Fraud Detection: Classification of Skewed Data’. ACM SIGKDD Explorations: Special Issue on Imbalanced Data Sets, 6; 50-59.

Quinlan, J. R. (1986). Induction of Decision Trees, in Machine Learning, Volume 1, pages 81-106.

Roberds, W. (1998). The impact of fraud on new methods of retail payment, Federal Reserve Bank of Atlanta Economic Review, First Quarter 42–52.

Sahin, Y. & Duman, E. (2011). “Detecting Credit Card Fraud by Decision Trees and Support Vector Machines” Proceedings of IMECS 2011, pp. 442-447, Hong Kong, March 16-18.

Seeja. K. R., Masoumeh, Z. (2014). Fraud Miner: A novel credit card fraud detection model based on frequent item set mining, The Scientific World Journal, Volume 2014, 252797

Statistics for General and Online Card Fraud, 20 June, 2007. <http://epaynews.com/statistics/fraud.html>.

Syeda, M., Zhang, Y. Q., & Pan, Y. (2002). “Parallel Granular Networks for Fast Credit Card Fraud Detection,” Proc. IEEE Int’l Conf. Fuzzy Systems, pp. 572-577.

Vatsa, V., Sural, S., & Majumdar, A. K., (2005) “A Game-theoretic Approach to Credit Card Fraud Detection,” Proc. First Int’l Conf. Information Systems Security, pp. 263-276.

Wayne, I. & Pat, L. (1992); Induction of one level Decision Trees in ML92: Proceedings of the Ninth International Conference on Machine Learning, Aberdeen, Scotland, 1–3 July 1992, San Francisco, CA: Morgan Kaufmann, pp. 233–240

Wilcoxon, F. & Wilcox, R. A. (1964). Some Rapid Approximate Statistical Procedures (Pearl River, NY: Lenderle laboratories)