ESANN'2005 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 27-29 April 2005, d-side publi., ISBN 2-930307-05-6.

UNSUPERVISED FUZZY ENSEMBLES AND THEIR USE IN INTRUSION DETECTION

Paul F. Evangelista[1], Piero Bonissone[2], Mark J. Embrechts[3],
and Boleslaw K. Szymanski[4]

Rensselaer Polytechnic Institute
(1,3) Department of Decision Sciences and Engineering Systems
(4) Department of Computer Science
Troy, New York 12180 - United States of America
(2) GE Global Research
Niskayuna, NY 12309- United States of America

Abstract. This paper proposes a novel method for unsupervised ensembles that specifically addresses unbalanced, unsupervised, binary classification problems. Unsupervised learning often experiences the curse of dimensionality, however subspace modeling can overcome this problem. For each subspace created, the classifier produces a decision value. The aggregation of the decision values occurs through the use of fuzzy logic, creating the fuzzy ROC curve. The one-class SVM is utilized for unsupervised classification. The primary source of data for this research is a host based computer intrusion detection dataset.

1 Introduction

The purpose of this paper is to illustrate synergistic combinations of multiple classifiers for the unbalanced, unsupervised binary classification problem. The data explored in this paper is commonly referred to as the Schonlau et. al. or SEA dataset[5]. Although this is a host based computer intrusion detection dataset, the applications of this work extend beyond computer intrusion detection.

Multiple Classifier Systems (MCS) is an active area of research today. A particularly interesting problem that involves MCS is the unbalanced, unsupervised binary classification problem. An unsupervised classifier, viewed in terms of novelty detection, cannot learn from true positive (intruders) examples. The only examples available to learn from are true negatives, or non-intruders. Furthermore, the problem is unbalanced with a low frequency of intruders. Given a classification problem of many dimensions (perhaps >10 variables), initial experimental results indicate that the creation of subspaces and aggregation of the subspace classification decision values results in improved classification over a model that utilizes all variables at once.

The dataset explored in this paper contains 5000 records that measure UNIX command usage from known users (50 users, each user contributes 100 records where a record is 100 UNIX shell commands). For every record, 26 variables have been created to measure the command usage behavior(such as frequency, popularity, commands never used; see [6, 13] for a description of variables). The data are evenly split with 2500 observations for training and 2500 observations for testing. After eliminating all positive cases from the training data, 2391 negative cases remain which are used for training the one-class SVM. In the testing data, there are 122 positive cases out of the 2500 observations. All data are normalized, subtracting the mean and dividing by the standard deviation unless indicated otherwise.

Schonlau et. al. [5] conducted the original work with this data to include: Bayes one-step Markov model, hybrid multistep Markov model, text compression, Incremental Probabilistic Action Modeling (IPAM), sequence matching, and a uniqueness algorithm[5]. Schonlau stressed the importance of minimizing false positives, setting a goal of 1% or less for all of his classification techniques. Schonlau's uniqueness algorithm, explained in [5], achieved a 40% true positive rating before crossing the 1% false positive boundary. Wang [14] used one-class training based on data representative of only one user and demonstrated that it worked as well as multi-class training. Coull et. al. [4] applied bioinformatics matching algorithm for a semi-global alignment to this problem. Lee [8] built a data mining framework for constructing features and model for intrusion detection. Szymanski and Zhang applied a recursive data mining algorithm for frequent patterns to detect intruders [13]. Evangelista et. al. [6] applied supervised learning through Kernel Partial Least Squares to the SEA dataset. Maxion contributed insightful work with this data that challenged both the design of the data set and previous techniques used on this data [9, 10].

2 Method

2.1 One-Class SVM

The one-class SVM is an outlier detection technique originally proposed in [11]. Stolfo and Wang [12] successfully apply the one-class SVM to this dataset and compare it with several of the techniques mentioned above. Chen uses the one-class SVM for image retrieval[3]. The simplest way to express the one-class SVM is to envision a sphere or ball, and the object is to squeeze all of the training data into the tightest ball feasible. Consider the following formulation of the one-class SVM originally from [11] and also clearly explained in [3]:

Consider $X_1, X_2, ..., X_n \in \chi$ instances of training observations, and Φ is a mapping into the feature space, F, from χ.

$$\min_{R\in\mathbb{R},\zeta\in\mathbb{R}^n,c\in F} \quad R^2 + \frac{1}{vn}\sum_i \zeta_i \tag{1}$$

$$\text{subject to} \qquad \| \Phi(X_i) - c \|^2 \leq R^2 + \zeta_i, \quad \zeta_i \geq 0 \text{ for } i \in [n]$$

This minimization function attempts to squeeze R, which can be thought of as the radius of a ball, as small as possible in order to fit all of the training samples. If a training sample will not fit, ζ_i is a slack variable to allow for this. A free parameter, v, enables the modeler to adjust the impact of the slack variables. The output, or decision value for a one-class SVM, takes on values generally ranging from -1 to +1, where values close to +1 indicate datapoints that fit into the ball and values of -1 indicate datapoints lying outside of the ball. All experiments in this paper utilize a linear kernel.

It is commonly understood that high dimensional data suffers from a curse of dimensionality. This curse of dimensionality involves the inability to distinguish distances between points because as dimensionality increases, volume grows exponentially and every point tends to become equidistant. This same curse of dimensionality occurs in the one-class SVM.

2.2 Subspace Modeling

The following approach attempts to overcome this curse of dimensionality. The technique involves creating subspaces of the variables and aggregating the outputs of the one-class SVM for each of these subspaces.

Intelligent subspace modeling is an important first step. Orthogonal subspaces are desired, because orthogonal subspaces measure different aspects of the data. The idea of creating diverse classifiers is not novel [1, 7], however in the literature the measures of classifier diversity involve functions of the classifier output. This is feasible with supervised learning, however in unsupervised learning this is more difficult because there are no true positive examples to measure diversity against. Intelligent subspace modeling measures diversity through the actual data. The method involves an analysis of the correlation between principal components of each subspace. This is by no means the only measure for subspace diversity, however this measure achieved promising results.

Given a scaled data matrix $\mathbf{X}$, containing m variables that measure n observations, create l mutually exclusive subspaces from the m variables. Assume there are k variables in every subspace if m is divisible by l. For each subspace, principal components can be calculated. The matrix that contains the principal component loading vectors (eigenvectors) will be referred to as $\mathbf{L}$. To determine correlation between principal components, calculate the principal component scores for each subspace, where $\mathbf{S}$=$\mathbf{XL}$. Let π_i represent subspace i, and consider $\mathbf{S}_i$ as the score matrix for the π_i. Calculate the pairwise comparison for every column vector in $\mathbf{S}_i$ against every column vector in $\mathbf{S}_j$, $i \neq j$. This would be the equivalent of concatenating $\mathbf{S}_i$ for all i and calculating the correlation matrix, Σ. Minimizing pairwise correlation across subspaces is the interest(principal components within subspaces are orthogonal and therefore their correlation is zero). However, there are a combinatoric number of subspace combinations to explore.

Searching for subspaces involved the implementation of a simple genetic algorithm, utilizing a chromosome with m distinct integer elements representing each variable. There are many possible objective functions that could pursue minimizing principal component correlation between subspaces. One approach is the following letting $q \in (1, 2, ..., l)$:

$$\min \max_{\forall \pi_q} \mid \rho_{ij} \mid \quad \forall (i \neq j) \tag{2}$$

The fitness of each member is simply the maximum $\mid \rho_{ij} \mid$ value from the correlation matrix such that ρ_{ij} measures two principal components that are not in the same subspace.

2.3 Output Processing

Classifier fusion techniques similar to the methods in this paper have been discussed in [1, 7]. Classifier fusion is a relatively new field and it is often criticized for lack of theoretical framework and too many heuristics [7]. We do not claim to provide a solution to this criticism. The method of classifier fusion in this paper is a blend of techniques from fuzzy logic and classifier fusion, and although it may be considered another heuristic, it is operational and should generalize to other security problems.

For each observation within each subspace selected, the classifier will produce a decision value, d_{ij}, where d_{ij} represents the decision value from the j^{th} classifier for the i^{th} observation. Since the distribution of the output from almost any classification technique is questionable, first consider a nonparametric measure for the decision value, or a simple ranking. o_{ij} represents the ordinal position of d_{ij} (for the same classifier, meaning j remains constant). For example, if d_{71} is the smallest value for the 1^{st} classifier, $o_{71} = 1$. This nonparametric measure allows comparison of classifiers without considering the distribution. However, the distribution is not ruled out altogether. p_{ij}, which is the scaled value for d_{ij}, is included. In order to incorporate fuzzy logic, o_{ij} and p_{ij} must be mapped into a new space of real numbers, let us call Λ, where $\Lambda \in (0, 1)$. This mapping will be $p_{ij} \rightarrow \delta_{ij}$ and $o_{ij} \rightarrow \theta_{ij}$ such that $\delta_{ij}, \theta_{ij} \in \Lambda$. For $o_{ij} \rightarrow \theta_{ij}$ this is a simple scaling procedure where all o_{ij} are divided by the number of observations, m, such that $\theta_{ij} = o_{ij}/m$. For $p_{ij} \rightarrow \delta_{ij}$, all p_{ij} values < -1 become -1, all p_{ij} values > 1 become 1, and from this point $\delta_{ij} = (p_{ij} + 1)/2$.

2.4 Fuzzy Logic and Decisions with Contention

There are now twice as many decision values for every observation as there were numbers of classifiers. Utilizing fuzzy logic theory, T-conorms and T-norms can be considered for fusion. The choice between T-norms and T-conorms depends upon the risk aversion of the decision maker and the nature of the classifier (recall that for the one-class SVM, intruders should be the most negative numbers). Caution against false negatives requires operating in the realm of the T-norms, creating more false alarms but missing fewer true positives. Caution against false positives requires T-conorms, perhaps missing a few true positives but generating fewer false alarms. Figure 1 illustrates the domain of aggregation operators.

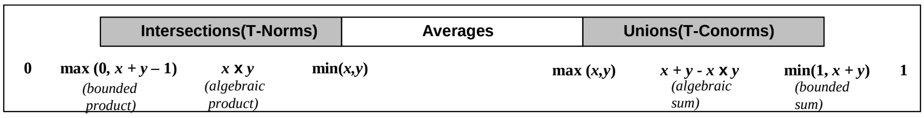
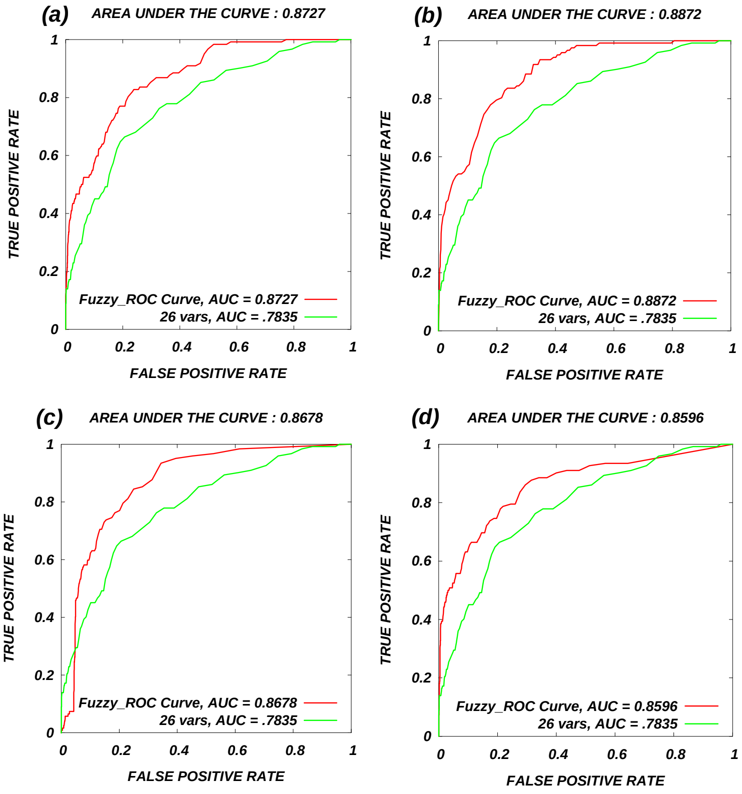


Fig. 1: Aggregation Operators

One problem with T-norms and T-conorms is that contention within aggregation is not captured. Contention refers to a vast difference of decision values between classifiers. However, contention can be captured and considered appropriately. There are numerous ways to measure contention, and one of the simplest is to consider the difference between the max and min decision values. If this difference exceeds a threshold, contention exists and it may be best to choose a different aggregator or make a cautious decision.

3 Results with Masquerading Data

Experimental results involved the SEA dataset that was mentioned earlier. There are m=26 variables and n=2500 observations in the training data. There are l=3 subspaces creating subspaces containing 9, 9, and 8 variables respectively. For each subspace we consider three principal components. The genetic algorithm used the fitness function shown in Equation 2, roulette wheel selection, a crossover rate of .6 and mutation rate of .01. The number of generations = population size = 50. The best subspaces achieved the following results: $\max_{\forall \pi_q} \mid \rho_{ij} \mid = .4 \; \forall (i \neq j)$. Given this subspace configuration, we utilized LIBSVM to calculate the one-class SVM decision variables [2].



ROC Plot	**Decision Rule for Each Observation (i); (t = threshold for contention)**
(a)	$max(\delta_{ij})\forall j$
(b)	$max(\theta_{ij})\forall j$
(c)	$max(\delta_{ij}, \theta_{ij})\forall j$
(d)	if $t < .5$, $max(\delta_{ij}, \theta_{ij})\forall j$; if $t \geq .5$, median $(\delta_{ij}, \theta_{ij})\forall j$

Fig. 2: ROC plots and associated decision rules

The decision rule was to take the maximum value unless there was contention $> .5$, and in this case we take the median of all decision values. The ROC curves shown in figure 2 illustrate the results.

4 Conclusions

This paper discusses a framework for a difficult domain of decision making: the unsupervised, unbalanced, binary classification problem with high dimensionality. It is common to encounter this domain in both the medical community and the security community. However, different risk aversion creates different policies for decisions. This framework capitalizes on theory from multivariate statistics, optimization, and information theory to present an approach for decision making and creation of such policies. The goal of the research discussed in this paper is to improve our ability to find synergistic combinations of classifiers measured by the fuzzy ROC curve. Future work includes finding alternate approaches for finding optimal orthogonal subspaces.

Acknowledgments

The authors would like to acknowledge Yongqiang Zhang for sharing his data and providing insightful discussions.

References

[1] Piero Bonissone, Kai Goebel, and Weizhong Yan. Classifier Fusion using Triangular Norms. Cagliari, Italy, June 2004. Proceedings of Multiple Classifier Systems (MCS) 2004.

[2] Chih Chung Chang and Chih-Jen Lin. LIBSVM: A Library for Support Vector Machines. http://www.scie.ntu.edu.tw/ cjlin/libsvm, Accessed 5 September, 2004.

[3] Yunqiang Chen, Xiang Zhou, and Thomas S. Huang. One-Class SVM for Learning in Image Retrieval. Thessaloniki, Greece, 2001. Proceedings of IEEE International Conference on Image Processing.

[4] Scott Coull, Joel Branch, Eric Breimer, and Boleslaw K. Szymanski. Intrusion Detection: A Bioinformatics Approach. Las Vegas, Nevada, December 2003. Proceedings of the 19th Annual Computer Security Applications Conference.

[5] William DuMouchel, Wen Hua Ju, Alan F. Karr, Matthius Schonlau, Martin Theus, and Yehuda Vardi. Computer Intrusion: Detecting Masquerades. *Statistical Science*, 16(1):1–17, 2001.

[6] Paul F. Evangelista, Mark J. Embrechts, and Boleslaw K. Szymanski. Computer Intrusion Detection Through Predictive Models. pages 489–494, St. Louis, Missouri, November 2004. Smart Engineering System Design: Neural Networks, Fuzzy Logic, Evolutionary Programming, Data Mining and Complex Systems.

[7] Ludmila I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms.* John Wiley and Sons, Inc., 2004.

[8] Wenke Lee and Salvatore J. Stolfo. A Framework for Constructing Features and Models for Intrusion Detection Systems. *ACM Transactions on Information and System Security (TISSEC)*, 3(4):227–261, 2000.

[9] Roy A. Maxion. Masquerade Detection Using Enriched Command Lines. San Francisco, CA, June 2003. International Conference on Dependable Systems and Networks.

[10] Roy A. Maxion and Tahlia N. Townsend. Masquerade Detection Using Truncated Command Lines. Washington, D.C., June 2002. International Conference on Dependable Systems and Networks.

[11] Bernhard Scholkopf, John C. Platt, John Shawe Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the Support of a High Dimensional Distribution. *Neural Computation*, 13:1443–1471, 2001.

[12] Salvatore Stolfo and Ke Wang. One Class Training for Masquerade Detection. Florida, 19 November 2003. 3rd IEEE Conference Data Mining Workshop on Data Mining for Computer Security.

[13] Boleslaw K. Szymanski and Yongqiang Zhang. Recursive Data Mining for Masquerade Detection and Author Identification. West Point, NY, 9-11 June 2004. 3rd Annual IEEE Information Assurance Workshop.

[14] Geoffrey I. Webb and Zijan Zheng. Multi-Strategy Ensemble Learning: Reducing Error by Combining Ensemble Learning Techniques. *IEEE Transactions on Knowledge and Data Engineering*, 16(8):980–991, 2004.