8-2005

# A Comparison of Generalizability for Anomaly Detection

Gilbert L. Peterson

Robert F. Mills

Brent T. McBride

Wesley T. Allred

# A Comparison of Generalizability for Anomaly Detection

Gilbert L. Peterson Robert F. Mills Brent T. McBride Wesley C. Allred
Department of Electrical and Computer Engineering
Air Force Institute of Technology
{gilbert.peterson, robert.mills}@afit.edu

## ABSTRACT
In security-related areas there is concern over the novel "zero-day" attack that penetrates system defenses and wreaks havoc. The best methods for countering these threats are recognizing "non-self" as in an Artificial Immune System or recognizing "self" through clustering. For either case, the concern remains that something that looks similar to self could be missed. Given this situation one could logically assume that a tighter fit to self rather than generalizability is important for false positive reduction in this type of learning problem.

This article shows that a tight fit, although important, does not supersede having some model generality. This is shown using three systems. The first two use sphere and ellipsoid clusters with a *k*-means algorithm modified to work on the one-class/blind classification problem. The third is based on wrapping the self points with a multidimensional convex hull (polytope) algorithm capable of learning disjunctive concepts via a thresholding constant. All three of these algorithms are tested on an intrusion detection problem and a steganalysis problem with results exceeding published results using an Artificial Immune System.

## Categories and Subject Descriptors
I.5.3 [Pattern Recognition]: Classifier design and evaluation, feature evaluation and selection.

## General Terms
Algorithms, Security

## Keywords
Anomaly detection, clustering, intrusion detection systems.

## 1. INTRODUCTION
The development of computer and network intrusion detection systems has been conducted along two paths. The first development thrust identifies signature elements of attacks, and includes them in an attack database. The database is then compared with incoming samples looking for matches, and if a match occurs, the user, packet, or file is blocked from the internal network. This is the approach taken by the majority of commercial intrusion detection and steganalysis products, with the capability of catching most known attacks with very few false alarms. A limitation of this approach is that the attack must be known before it can be given a signature and blocked. Subtle, stealthy probes will most likely not be picked up by this type of system (Williams et al, 2001). Additionally, due to the sample arrival rate and database matching procedure, the speed at which attacks can be blocked will be limited.

An alternative attack matching method is based on anomaly detection. In this approach, a machine learning algorithm learns a model of normal operating behavior so that abnormal conditions can be identified. The advantage of this approach is that novel attacks (for which signatures have not been identified) may be identified and blocked. Additionally, the approach may be much quicker, because maintenance of an online signature database for matching purposes is not required. A disadvantage is that an attacker with knowledge of which attributes are used for detection could construct stealthy attacks that avoid using or manipulate the attributes used by the machine learning algorithm to appear normal.

In order to detect attacks from an attacker trying to blend in to normal traffic, we examine fitting the normal "self" data more closely. Figure 1 shows the results of applying the modified *k*-means sphere, ellipse, and the convex polytope algorithms to each class separately for a simple two class problem. As can be seen from just this simple example, the generalizability of the model decreases as the model improves its tightness to the data points. One could also imagine that if these classes were more interspersed that the convex polytope which provides the closest fit to the data would perform the best. Given a domain in which the attackers attempt to craft an attack that appears as close to normal (self) as possible, a learning approach which fits the model closely could be seen as important.

In the following sections we discuss related work on anomaly detection for the intrusion detection and steganalysis domains used for testing. This is followed by a discussion of how we have modified *k*-means and the thresholding element required for the convex polytope to learn disjunctive concepts. The test results are then presented showing that a tight fit is important but that generalizability is still necessary given the sampling of the normal/self space.

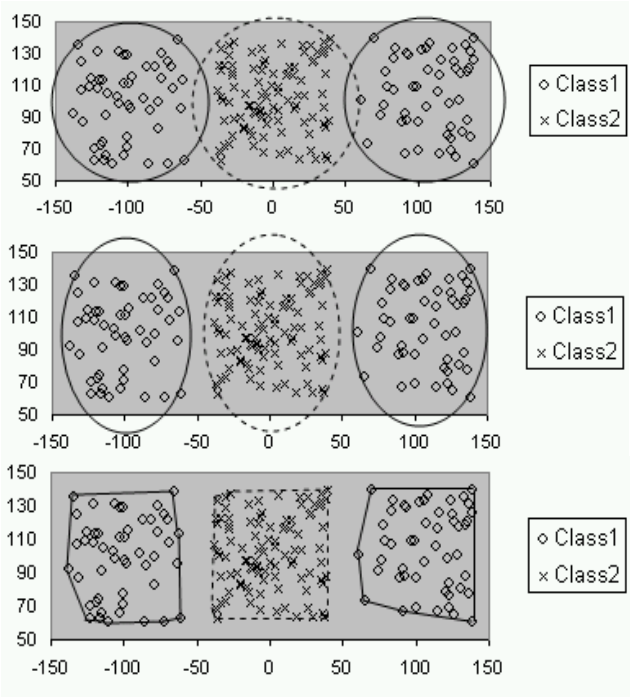


**Figure 1. The 2-Class Problem with Sphere, Ellipse and Convex Polytope**

## 2. RELATED WORK

In this section we discuss related work on anomaly detection for the intrusion detection and steganalysis domains.

### 2.1 Intrusion Detection Systems

Anomaly detection systems have been built making use of rule learning, neural networks, Artificial Immune Systems (AIS), and clustering methods. The clustering methods and Artificial Immune Systems are most closely related to this work in that the systems can be trained using only normal traffic. Artificial Immune Systems train on normal data by enclosing non-self space with randomly generated immune system cells. These cells then take part in an evolutionary algorithm evolution process until as much of the non-self space as possible is covered with none of the cells impinging on self space (Harmer, et al, 2002). We compare our results with an AIS technique (Dasgupta and Gonzales, 2002) in a later section. Researchers have also made use of system call activity as another source of data for anomaly based intrusion detection (Hofmeyr, et al, 1998; Nguyen, et al, 2003; and Tan, et al, 2003).

The application of clustering to intrusion detection groups network traffic into subclasses such that the members in one subclass are similar, while members of different subclasses are distinct. Several techniques have been studied, such as *k*-means, Self Organizing Maps (SOM), Neural-Gas, and Mixture-of-Spherical Gaussians (MOSG) to name a few. Clustering has been shown to produce very good results as an unsupervised IDS technique (Zhong, et al, 2004) and for data reduction prior to categorization (Zanero and Savaresi, 2004). In addition, there is a variation of *k*-means that also contains a stochastic element which behaves like an AIS (Guan, Ghorbani and Belacel, 2003).

### 2.2 Steganography

Steganography refers to hiding information in an innocuous place so that it may be transmitted without notice. In the digital realm, specifically digital images, the message is hidden within a cover image. The hiding or steganograpy process varies the image's pixels in such a way that the changes are virtually undetectable to the human eye. The cover images that provide the most difficulty for message detection are JPEG images.

JPEG compression is a lossy image compression technique that exploits the fact that the eye cannot detect small changes in an image. In a JPEG image, a message is stored using the least significant bit (LSB) or even through rounding errors on the quantized discrete cosine transform (DCT) coefficients representing 8x8 blocks of the image.

For the lossy steganography problem there have only been a few applications of learning models for normal images, and none have used any type of clustering. Approaches which make use of both self and non-self data have used Fisher's linear discriminant, Support Vector Machines with image quality metrics, and wavelet statistics calculated from the suspect images (Farid and Lyu, 2002; Lyu and Farid, 2002; and Avcibas, et al, 2002). A survey of the metrics available and their utility is provided in (Kharazzi, et al, 2004).

Blind or one-class learning methodologies have consisted of Artificial Immune Systems (Jackson, 2003) and single class Support Vector Machines (Lyu and Farid, 2004).

## 3. METHODS

In this section, we discuss how we have modified *k*-means and the thresholding element required for the convex polytope to learn disjunctive concepts.

### 3.1 *k*-means

The *k*-means algorithm is a clustering algorithm which assigns points to clusters by attempting to minimize the sum of squared errors within groups, or the sum of the distance squared between each point and the centroid of its assigned cluster. The algorithm then iteratively updates the cluster centroids moving the centroid toward the center of the cluster's points. This is followed by reassigning points to different clusters until it can no longer reduce the sum of squared within group errors. The time complexity of the *k*-means algorithm is O(*knr*) for *k* clusters, *n* points, and *r* iterations (Wong, Chen and Yeh, 2000).

As *k*-means is being used as a classifying algorithm, a class is described by a set *S* of *k* hyper-spheres. First, the *k*-means clustering algorithm partitions the self data into *k* different clusters, where *k* acts as a tolerance parameter for the hyper-sphere classification algorithm by controlling the partitioning of the self data. For the spherical version, a radius for each cluster is calculated from the distance between the corresponding centroid and the most distant point in the cluster. A new sample is declared part of self if it falls within one of the cluster radii.

A good IDS or steganalysis detection system should have a high probability of detection ($P_D$) and small probability of false alarm ($P_F$). The challenge is finding the appropriate balance between these opposing objectives. For example, decreasing the volume

of the training class reduces the number of missed detections, thereby improving $P_D$, but at the expense of more false alarms and a higher $P_F$. As a method to create a tradeoff between $P_D$ and $P_F$, a tolerance parameter, $0 < \delta \leq 1$, applied to each cluster's radius provides a simple method to constrain the clusters from covering too much non-self space.

An ellipsoid model was also used to strike a balance between the loose fitting spherical *k*-means representation of self space and the very tight fitting convex polytope described in the next section. An ellipsoid in *d* dimensions is represented by three parameters defining its location size (*s*: a scalar value), ($\mu$: a *d*-vector specifying the center point), and shape ($\Sigma^{-1}$: a *d*-by-*d* matrix describing the shape of the ellipsoid). Any point *x* on the ellipsoid boundary (locus) satisfies

$$(x-\mu)^T \sum^{-1} (x-\mu) = s$$

The ellipsoid model in *k*-means minimizes to find not only a cluster center $\mu$ but the shape $\Sigma^{-1}$ as well. This increases the creation time complexity to $O(kn^2d^2)$.

## 3.2 Convex Polytope

A *d*-polytope is a closed geometric construct bounded by the intersection of a finite set of hyperplanes, or halfspaces, in *d* dimensions (Coxeter, 1973). The polytope is convex if all points in a line segment between any two points on the polytope boundary lie either within the polytope or on its boundary. A convex hull of a set of points *S* in *d* dimensions is the smallest convex *d*-polytope that encloses *S* (O'Rourke, 1998). Each vertex of this enclosing polytope is a point in *S*.

For classification purposes, the convex polytope for a class *C* is built from the set *T* of *d*-vectors from the sample space. If the desired geometric shape is a convex *d*-polytope, then the convex hull *H* of *T* is computed. There are several algorithms for computing convex hulls in higher dimensions (Avis, et al, 1997). This research uses the *qhull* program (Barber, et al. 1997), which has a time complexity of $O(n^{\lfloor d/2 \rfloor})$ for *n* points in *d*-space. A distinct test point *p* is declared to be a match (member of class *C*) if and only if it is bounded by the polytope defined by *H*.

To account for class disjunction, we define $0 \leq \beta \leq 1$ as a tolerance parameter to control the creation of smaller convex hulls. With $\beta = 1$, the algorithm creates a single convex polytope around all training points. As $\beta$ decreases, the potential number of smaller polytopes increases, and their combined hyper-volume in the attribute space decreases. For the extreme case $\beta = 0$, no convex hull models are created and all test points are subsequently rejected. The method for constructing the smaller convex hulls is described in (McBride and Peterson 2004).

Selecting different values of $\beta$ allows us to achieve the desired balance between false positive and false negative error probabilities. If instances of all possible testing classes are available when creating the class model, then the value of $\beta$ that best fits the training data (i.e., provides an appropriate balance between false positives and false negatives) can be found through experimentation.

# 4. TESTING

The flexibility of these classifiers allows for uses in many possible domains. Our research focuses on evaluating anomaly classification as applied to the problems of detecting suspicious computer network activity and steganography, both of which may accompany an attack against a computer network by an outsider. These domains also show the classification capabilities on windowed time series data (IDS) as well as discrete sampled data (steganalysis).

## 4.1 IDS Experiment

The dataset used for this experiment was obtained from the Lincoln Laboratory of the Massachusetts Institute of Technology. MIT maintains data sets with normal and abnormal information collected in a test network (Haines, et al, 1999). Although this data set has been shown to be statistically different from normal traffic (Mahoney and Chan, 2003), its many uses by the research community allow for comparison with other approaches. For this experiment, we used the 1999 data set, with week 1 (normal traffic) to train our classifiers, and week 2 (normal traffic mixed with attacks) for testing. Abnormal activity includes both internal (misuse) and external (hacking or denial of service) attacks, but not the external use of operating system or application exploits, as shown in Table 1.

**Table 1. Week 2 Attack Profile**

| Day | Attack | Attack Type | Start Time | Duration |
|---|---|---|---|---|
| 1 | Back | DOS | 9:39:16 | 00:59 |
| 2 | Portsweep | Probe | 8:44:17 | 26:56 |
| 3 | SATAN | Probe | 12:02:13 | 2:29 |
| 4 | Portsweep | Probe | 10:50:11 | 17:29 |
| 5 | Neptune | DOS | 11:20:15 | 04:00 |

We follow the same data preparation methodology as (Dasgupta and Gonzalez 2002) and collect statistics on the number of bytes per second, number of packets per second, and number of Internet Control Management Protocol (ICMP) packets per second for classification features. These features were sampled each minute from the raw *tcpdump* data files. Dasgupta and Gonzalez showed that while none of these features alone could reliably detect the five attacks, combining the features was quite effective. They also explored overlapping the time series as a means of detecting temporal patterns, with their best results generated using a sliding window of three seconds.

False positive and true positive probabilities were calculated by comparing the classifier output with the Week 2 attack data. Table 2 shows the results of testing the *k*-means sphere and ellipse classifiers, the convex polytope, and the AIS results (Dasgupta and Gonzalez, 2002) on the MIT IDS dataset. Multiple tests for each algorithm were run, and the table contains the best results found for $P_F$ and $P_D$ of each algorithm with the exception of the AIS which includes the results for 1 and 3 time slices from (Dasgupta and Gonzales, 2002).

**Table 2. IDS Results**

| | Sphere | | Ellipse | | Polytope | | AIS | |
|---|---|---|---|---|---|---|---|---|
| | k=75 δ = 1.0 | k=100 δ = 0.9 | k=30 δ = 1.0 | k=75 δ = 1.0 | $\beta > 0.3$ | $\beta = 0.1$ | 1 time slice | 3 time slices |
| $P_D$ (%) | 1.82 | 5.45 | 98.2 | 100.0 | 98.2 | 100.0 | 92.8 | 98.0 |
| $P_F$ (%) | 0.0 | 1.02 | 0.0 | 0.2 | 0.27 | 0.35 | 1.0 | 2.0 |

During testing of the *k*-means variations, *k*-values ranged from 1-100 in steps of 5 and δ=0.9, 0.95, and 1.0 were used to determine classifier sensitivity as a function of the number of ellipsoids used to fit the training data. As shown, the ellipsoid model with its added capability of generalizing beyond the strict sampling is able to better fit the training data over the convex polytope which was trained using several values of $\beta$ for $0 \leq \beta \leq 1$. In addition, the results show that the sphere version of *k*-means performs very poorly predominantly because it inaccurately covers the training attribute space by also enclosing space including anomalous data points. This continues even as *k* increases and each cluster decreases in size. The reason the sphere does not perform as well as the other two geometric constructs is that the *k*-means classifier uses the point furthest from the mean for each cluster to estimate the size of the hyper-sphere, resulting in an over-generalization. This contrasts with the ellipse and convex polytopes which try to maintain a closer fit to the training data.

These results imply that the convex polytope and the ellipse *k*-means had little trouble fitting the training data, and that their ability to more tightly fit the self space improves their overall performance for classification based on these three statistical attributes. Additionally this shows that although both models fit the data closely that the added generality of the ellipse *k*-means assists in reducing the false positives which is counter to the assumption that one would want the closest fit to the training data for anomaly detection.

## 4.2 Steganalysis Experiment

For this domain we test using the wavelet coefficient statistics (Farid and Lyu, 2003) derived from a database of 1,100 grayscale images. The best three of the 36 coefficients determined by J-score are extracted from each image. In addition to clean images, the testing set includes steganographic images created with Jsteg, and Outguess with and without statistical correction. For each of these three steganography methods, images are created using 100%, 50%, 25%, and 12.5% of the cover image's embedding capacity.

Figure 2 shows the results from the steganography testing compared with the results using the same testing domain and an AIS as the classifier from (Jackson 2003). As seen with the IDS problem, the closer fit to the self space provided by both the convex polytope and ellipse *k*-means outperforms the more general sphere *k*-means. However, it is also shown that striving for the closest fit possible, i.e. the convex polytope, is also not the direction that should be pursued. Specifically, the lack of generality, especially on the Jsteg dataset, is detrimental to the convex polytope over the ellipse *k*-means.

# 5. CONCLUSIONS

For security anomaly detection domains, a concern prior to fielding the system is whether it can be spoofed by an attacker manipulating their attack to appear similar to normal traffic. In order to combat such an event we proposed that a model of self should fit the normal self sample tightly. This theory has been tested on two security domains, namely intrusion detection and steganalysis.

This paper shows that while the convex polytope algorithm provides the tightest fit to self, the ellipsoid *k*-means provides the best balance between a tight fit and sufficient generality. The small amount of generality provided by the ellipse resulted in a better ability to detect novel events that may otherwise go undetected in a classifier with a tight fit. This is especially worrisome in a network intrusion scenario in which the attack pattern appears as close to normal as possible. The results have demonstrated that a tight fit is important but does not obviate the need for generality.

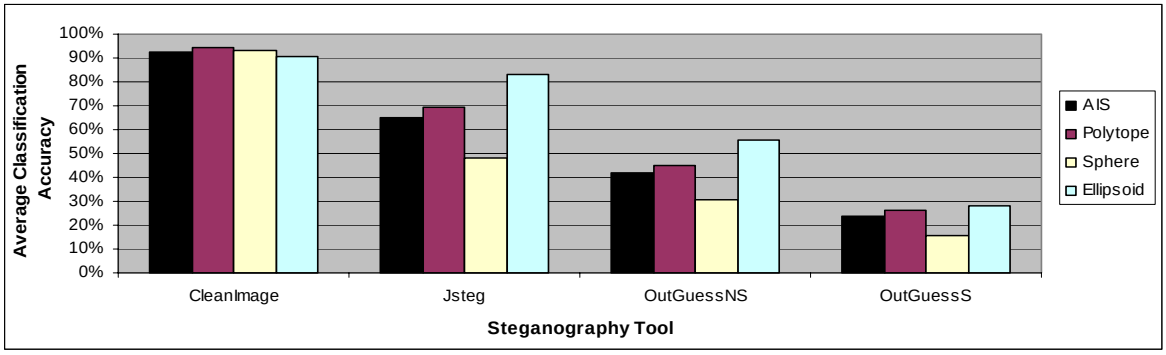


**Figure 2. Steganography Results**

## 6. REFERENCES


[1] Avcibas, I., Memon, N., and Sankur, B. "Image Steganalysis With Binary Similarity Measures", *International Conference on Image Processing*, Rochester, NY, September 2002.

[2] Avis, D., Bremner, D., and Seidel, R. "How Good are Convex Hull Algorithms?" *ACM Symposium on Computational Geometry*, Nice, France, 1997.

[3] Barber, C., Dobkin, D., and Huhdanpaa, H. "The Quickhull Algorithm For Convex Hulls", ACM Trans. on Mathematical Software, 22, 469-483, 1997.

[4] Coxeter, H. S. M. *Regular Polytopes, 3rd ed.* New York: Dover, 1973.

[5] Dasgupta, D., and Gonzales, F. "An Immunity-Based Technique to Characterize Intrusions in Computer Networks", *IEEE Trans. on Evolutionary Computation, Vol 6*, June 2002.

[6] Faird, H. and Lyu, S. "Higher-order Wavelet Statistics and their Application to Digital Forensics", *IEEE Workshop on Statistical Analysis in Computer Vision*, Madison, Wisconsin, June 2003.

[7] Guan, Y., Ghorbani, A., and Belacel, N. "Y-Means: A Clustering Method for Intrusion Detection", *IEEE Canadian Conference on Electrical and Computer Engineering CCECE*, Montréal, Canada, May 2003.

[8] Haines, J., Lippmann, R. , Fried, D., Tran, E., Boswell, S., and Zissman, M. "1999 DARPA Intrusion Detection System Evaluation: Design and Procedures", MIT Lincoln Laboratory Technical Report.

[9] Harmer, P., Williams, P., Gunsch, G., and Lamont, G. "An artificial immune system architecture for computer security applications", *IEEE Transactions on Evolutionary Computation, Vol 6*, June 2002.

[10] Hofmeyr, S., Forrest, S., and Somayaji, A., "Intrusion Detection Using Sequences of System Calls", *Journal of Computer Security, Vol. 6*, pp. 151-180 (1998).

[11] Jackson, J. *Targeting Covert Messages: A Unique Approach For Detecting Novel Steganography*, Masters Thesis, Air Force Institute of Technology, Wright Patterson Air Force Base, Ohio, 2003.

[12] J-Steg Steganograpy software for Windows, http://members.tripod.com/steganography/stego/software.html.

[13] Kharrazi, M., Sencar, T., and Memon, N. "Benchmarking Steganographic And Steganalysis Techniques", *EI SPIE San Jose*, CA, January 16-20, 2005.

[14] Lyu, S., and Farid, H. "Detecting Hidden Messages Using Higher-Order Statistics And Support Vector Machines," *Information Hiding: 5th International Workshop, IH 2002*, Noordwijkerhout, The Netherlands, October 7-9, 2002.

[15] Lyu, S., and Farid, H. "Steganalysis Using Color Wavelet Statistics And One-Class Support Vector Machines," *SPIE Symposium on Electronic Imaging*, San Jose, CA, 2004.

[16] McBride, B. and Peterson, G. "Blind Data Classification using Hyper-Dimensional Convex Polytopes", *Proceedings of the 17th International FLAIRS Conference*, Miami Beach, FL, 2004.

[17] Mahoney, M., and Chan, P., "An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection", *Proceedings of the Recent Advances in Intrusion Detection, RAID 2003*, Pittsburgh, PA, USA, September 8-10, 2003.

[18] Nguyen, N., Reiher, P., and Kuenning, G. "Detecting Insider Threats by Monitoring System Call Activity", *2003 IEEE Workshop on Information Assurance*, United States Military Academy, West Point, NY, June 2001.

[19] Provos, N., "Defending Against Statistical Steganalysis", *Proceedings of the 10th USENIX Security Symposium*, Washington, DC, 2001.

[20] O'Rourke, J. *Computational Geometry in C, 2nd ed.* Cambridge, England: Cambridge University Press, 1998.

[21] Tan, K., McHugh, J., and Killourhy, K. "Hiding Intrusions: From the Abnormal to the Normal and Beyond", *Information Hiding: 5th International Workshop, IH 2002*, Noordwijkerhout, The Netherlands, October 7-9, 2002.

[22] Williams, P., Anchor, K., Bebo, J., Gunsch, G., and Lamont, G. "Warthog: Towards a Computer Immune System for Detecting 'Low and Slow' Information System Attacks", *Proceedings of the Recent Advances in Intrusion Detection Symposium, RAID 2001*, Davis, California, 2001.

[23] Wong C., Chen, C., and Yeh, S. "K-Means-Based Fuzzy Classifier Design", *Proceedings of the Ninth IEEE International Conference on Fuzzy Systems, Vol. 1*, pp. 48-52, 2000.

[24] Zanero, S., and Savaresi, S. M. "Unsupervised Learning Techniques for an Intrusion Detection System", *Proceedings of the 19th Annual ACM Symposium on Applied Computing*, Nicosia, Cyprus, 2004.

[25] Zhong, S., Khoshgoftaar, T., and Seliya, N., "Clustering-Based Network Intrusion Detection", To appear in *International Journal of Reliability, Quality, and Safety Engineering (IJRQSE)*, 2005.