Proceedings of the 2001 IEEE
Workshop on Information Assurance and Security
United States Military Academy, West Point, NY, 5-6 June, 2001

Robustness of Canberra Metric in Computer Intrusion Detection

Syed Masum Emran and Nong Ye, *Member, IEEE*

***Abstract*--Detecting computer intrusions has become an increasingly potential research area in recent years. More and more attacks are conducted towards various kinds of information system. To integrate the research works in this area Defense Advanced Research Projects Agency (DARPA) has established a corpus for evaluating the different intrusion detection technique. In this study we have used the data set provided by the corpus which has been collected by the Massachusetts Institute of Technology (MIT) Lincoln Laboratory. Of the two types of intrusion detection techniques – signature recognition and anomaly detection -- we have developed a multivariate statistical-based anomaly detection technique, namely, Canberra technique. It does not suffer from the normality assumption of the data. The Canberra distance metric is used for similarity/dissimilarity comparison. We applied the metric in this research to find out which activity differs from the norm profile established during training and raise signal if the deviation is significant. We applied a small set of data containing both normal and attack events and set up three test cases – ideal, mixed and noisy – to determine its robustness. In the ideal test case all the attacks events are placed in together. In the mixed test case, the normal and attacks are evenly mixed. In the noisy test case small number of normal events are placed between the attack sessions. By plotting the performance of Canberra distance metric in these test cases through Receiver Operating Characteristics (ROC) curve we conclude that the technique perform very well only in ideal case. The paper also addresses the future direction of this research.**

***Index Terms*--Intrusion Detection; Canberra; Multivariate Statistical Process Control**

I. Introduction

With the rapid development of the Internet and e-commerce systems and increasing number of hackers in recent years network security has become a critical issue. An intrusion stems either from inside the network or outside the network and can steal classified information or create havoc in the network and halt normal user activities, incurring huge monetary and credibility losses. In case of internal attack the event activity log on the hosts, from which the attack has been performed, has to be analyzed to perceive the intrusive activities. External attacks can be detected by analyzing the pattern of network traffic data along with the activities in the victim hosts. In this paper we study the robustness of an intrusion detection techniques by applying host event activity logs. The paper is organized as follows. Section II describes some of the existing intrusion detection techniques. Section III summarizes the information about audit data used in this study. Section IV and V present the Canberra technique and experimentation used in testing its robustness respectively. Section VI analyzes the results. Section VII addresses the future direction. Section VIII summarizes the paper.

II. Intrusion Detection Techniques

Existing intrusion detection techniques can be classified into two categories - anomaly detection techniques and signature recognition (also coined as "misuse detection" in some literature) techniques [1] [2]. Signature recognition techniques maintain a database of known intrusions and try to match observed activities with the known attacks. Anomaly detection techniques establish a norm profile using normal activities and detect any deviation of observed activities from the norm profile. Signature detection technique is very good in detecting known attacks, whereas anomaly detection techniques can detect variants of known attacks and new attacks as well. Therefore, these two types of techniques complement each other.

Signature recognition techniques have been used in most of existing intrusion detection systems, including NSM/ASIM, NetRadar, IDES/NIDES, EMERALD, NetRanger, Stalker, CMDS, NetStalker, TCP Warpper, Tripwire, SATAN, and STAT [1]–[7]. Intrusion signatures have been characterized as strings, event sequences, activity graphs, and intrusion scenarios consisting of event sequences, their preconditions and target compromised states. Finite state machines [3], colored Petri Nets [4], associate rules [5] and production rules of expert systems [6] [7] have been used to represent and recognize intrusion signatures. Intrusion signatures are either manually encoded or automatically learned through data mining. The most significant drawbacks of signature-based approaches are: 1) it can detect only those attacks that it was trained to, 2) novel or event variants of common attacks often go undetected, 3) in a scenario where new kinds of attacks are detected very frequently, signature recognition techniques are not feasible.

Three types of anomaly detection techniques exist in use: string-based, specification-based, and statistical-based [8]–[14]. String-based anomaly detection techniques [13] [15] collect sequences of system calls or audit events that appears in normal activities, represent those sequences as strings and build norm profile, and employ either negative selection [13] or positive selection [15] to determine whether an observed string deviates from the string-based norm profile. Specification-based anomaly detection techniques [14] use predicates in formal logic to build the norm profile, and applies logical reasoning to determine the consistency of

ISBN 0-7803-9814-9/$10.00 © 2001 IEEE

observed activities with the norm profile. Statistical-based anomaly detection techniques use statistical properties of normal activities to build a norm profile, and employ statistical tests to determine whether observed activities deviate significantly from the norm profile. The drawbacks of anomaly detection techniques are: 1) well-known attacks may not be detected if they fit the established profile of the user, 2) A malicious user who knows that she is being profiled can change her profile slowly overtime to essentially train the anomaly detection method to learn her malicious behavior as normal, 3) a high false-positive rate may result for a narrowly trained detection algorithm, or a high false-negative rate may result for a broadly trained anomaly detection approach.

Statistical-based anomaly detection techniques are inherently capable of handling variations and noises involved in normal activities, which the string-based anomaly detection techniques and specification-based anomaly detection techniques lack. A norm profile must consider and represent variations of normal activities for distinguishing truly anomalous activities from expected variations of normal activities.

Most of the studies on statistical-based anomaly detection techniques [8]–[12] are based on a statistical technique developed for IDES/NIDES. Their technique computes test statistics (called Q statistic and S statistic) using data on a single measure. This technique is sensitive to the normality assumption, i.e., if data on a measure are not normally distributed, the technique yields a high false alarm rate. Moreover, the statistical norm profile is built for only one measure of activities in information systems and hence the technique is univariate. However, intrusions often affect multiple measures of activities collectively. Anomalies resulting from intrusions may cause deviations on multiple measures in a collective manner which their technique fails to recognize. To address the lack of multivariate analysis techniques for intrusion detection in existing work, we have developed a statistical-based multivariate analysis technique, namely, Canberra, and applied it towards intrusion detection and tested its performance in several scenarios.

III. DATA SOURCE

The Information Systems Technology Group of Massachusetts Institute of Technology (MIT) Lincoln Laboratory, under Defense Advanced Research Projects Agency (DARPA) Information Technology Office and Air Force Research Laboratory (AFRL) sponsorship has collected and distributed the network data containing normal and intrusive activities and developed a guideline for evaluating existing intrusion detection systems in 1998 [16]. They have created normal traffic similar to that on a government site containing 100's of users on 1000's of virtual hosts. There are three types of data - network traffic data, system level audit data and file system state data. We use only the system level audit data which is collected by the Solaris Basic Security Module (BSM).

The BSM audit data is provided into two parts - training data and testing data. The training data set consists of seven weeks of network-based attacks in the midst of normal background data. The testing data contains another two weeks of data which contain all those attacks that were in training data and also new types of attacks. The data records in testing data set are not labeled. The attacks were drawn roughly evenly from four general classes of attacks - surveillance, denial of service, user-to-root and remote-to-local. There are more than 300 instances of 32 different attacks arising from 7 attack scenarios Detail information about these attacks can be found in [17].

We build a small set of data in two ways - the normal events are adopted from the public section of the 1998 MIT Lincoln Laboratory data and the attack events are collected through attack simulation at the Information Systems and Assurance lab, ASU. There are 2316 normal events in the training data. There are 7 attack sessions in the testing data which are 215, 225, 54, 36, 413, 247 and 35 events in length respectively and arises from three types of attacks, namely, password guessing, suspicious program usage and port scanning. Since anomaly detection techniques are trained with normal events only, we do not include any attack events during training. We train the technique by scanning the training data twice. In the first scan we build the long term profile by computing the mean observation value and in the second scan we build an empirical distribution by computing the mean and standard deviation of the Canberra distance metric computed for each of the normal events. We have put 703 normal events and 1225 attack events in the testing data set. During testing we calculate the Canberra distance metric from the observation value and then use the mean and standard deviation of the Canberra distance metrics and compare with the upper limit to produce a signal.

We use the computer audit data collected from MIT Lincoln Laboratory. The Basic Security Module (BSM) of UNIX monitors each ongoing event in the computer system. There are 284 different types of events. In UNIX, there are thousands of commands available. But the audit events are more close to the core of the operating system, hence the event type is more representative than the actual command sequences used. For example, we can use any text editor, such as, vi, ed, pico to edit a file, but most of the time the audit event stream will contain the following event types: AUE_EXECVE, AUE_OPEN_R, AUE_ACCESS, AUE_STAT. We refer to this approach as "event type testing" [18] [19].

IV. CANBERRA TECHNIQUE

The main objective in our research is to group the incoming events into normal events or attack events by calculating intrusion warning (IW) values using some effective distance metrics. The distance metric value is used to find the similarity or dissimilarity of the current observation from the already established normal profile. The IW value tells us how far the observed activity is from the norm profile in a 0 to 1 scale. An IW value of 1 means that the observed activity belongs to an

attack sequence, a value of 0 indicates that it is normal. The values between 0 to 1 tell us the intrusiveness of the event, the higher the value, the higher the intrusiveness. This normalized IW value gives us more information than the raw distance metric value.

To find the distance between normal profile and current observation value, we can use many distance metrics. The Euclidean (straight-line) distance between two *k*-dimensional observations $\mathbf{x} = [x_1, x_2, \ldots, x_k]'$ and $\mathbf{y} = [y_1, y_2, \ldots, y_k]'$ is:

$$d(\mathbf{x},\mathbf{y}) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_k - y_k)^2} = \sqrt{(\mathbf{x}-\mathbf{y})'(\mathbf{x}-\mathbf{y})} \quad (1)$$

The statistical distance between the same two observations is of the form

$$d(\mathbf{x},\mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})'\mathbf{A}(\mathbf{x}-\mathbf{y})} \quad (2)$$

Ordinarily, $\mathbf{A} = \mathbf{S}^{-1}$, where $\mathbf{S}$ contains the sample variances and covariances.

Another distance measure is the Minkowski metric

$$d(\mathbf{x},\mathbf{y}) = \left[\sum_{i=1}^{k} |x_i - y_i|^m\right]^{1/m} \quad (3)$$

For $m = 1$, $d(\mathbf{x}, \mathbf{y})$ measures the "city-block" distance between two points in *k*-dimensions. For $m = 2$, $d(\mathbf{x}, \mathbf{y})$ becomes the Euclidean distance. In general, varying *m* changes the weight given to larger and smaller differences. Two additional popular measures of "distance" or dissimilarity are given by the Canberra metric and the Czekanowski coefficient. Both of these measures are defined for nonnegative variables only. The Canberra metric and the Czekanowski coefficient are given by equations (4) and (5) [20].

$$d(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{k} \frac{|x_i - y_i|}{(x_i + y_i)} \quad (4)$$

$$d(\mathbf{x},\mathbf{y}) = 1 - \frac{2\sum_{i=1}^{k} \min(x_i, y_i)}{\sum_{i=1}^{k} (x_i + y_i)} \quad (5)$$

We have applied only the Canberra metric in this research. In our case we want to measure how much the current observation value differs from the established mean observation value. The nominator in the Canberra metric equation signifies the difference and the denominator in fact normalizes the difference. We modify the Canberra metric equation in this way:

$$C = \sum_{\text{All dimensions}} \frac{|observed - expected|}{(observed + expected)} \quad (6)$$

V. EXPERIMENT

We need to convert the incoming event stream to a stream of observation values. During training we compute the expected value of the observation values. After that, we build empirical distribution for the Canberra technique by computing the distance metrics and their mean and standard deviation. We set upper control limit for using the mean and standard deviation. During testing we compute the observation values and the Canberra distance metric and then compare it with the upper control limit set during training stage. We signal an event as abnormal if its observation value exceeds the upper control limit; otherwise, if the observation value stays within the limit we consider the event as normal.

We have to use an efficient method to calculate the observation values from the system audit event stream as the quality of the statistical techniques depends on it. Since there are 284 types of events in BSM audit data, number of categories *k*, i.e., number of attributes or variables associated with each event is 284. An event can only be one of the event types. Therefore, the observation value is a 284-dimensional vector vector $\mathbf{O} = (O_1, O_2, \ldots O_{284})$. We can calculate the vector by counting the number of event types that appeared in the event stream so far.

We can use the exponentially weighted moving average (EWMA) technique for smoothing out the observation values. When we choose an appropriate smoothing constant λ, the observation measure reflects the "most recent past" characteristics of variables. We can use the following formulae for calculating the observation value of the n^{th} event in the event stream:

$$\mathbf{O}_n = (O_{1,n}, O_{2,n}, \ldots, O_{284,n}),\ n \geq 0 \quad (7)$$

$$O_{i,n} = \lambda \times \vartheta + (1 - \lambda) \times O_{i,n-1};$$

$$O_{i,0} = 0,\ 1 \leq i \leq k, \text{ and } k = 284 \quad (8)$$

In formula (7), $\mathbf{O}_n$ is the smoothed observation vector for the n^{th} event. The individual dimensional values are computed using formula (8) where, $O_{i,n}$ is the smoothed observation value for category (event type) *i* for the n^{th} event, θ is an indicator function which 1 if the category *i* is present in the current observation, 0 otherwise, $O_{i,n-1}$ is the previous smoothed observation, λ is the smoothing constant ($0 < \lambda < 1$). The smoothing constant is usually set to 0.3 [21]. With this "most recent past" approach, we add the time characteristic into the observation value. The observation value not only shows the current probability distribution of category vector, but also reflects the recent period probability distribution.

We have used $\overline{\mathbf{X}} = (\overline{X}_1, \overline{X}_2, \ldots, \overline{X}_{284})$ to calculate the expected value of each of the variables being tracked. It reflects a long period characteristic of the variables we observe. We use the following incremental formula:

$$\overline{X}_{(i,n)} = \frac{(n-1)\overline{X}_{(i,n-1)} + O_{(i,n)}}{n};$$

$$\overline{X}_{(i,0)} = 0,\ 1 \leq i \leq k,\ n \geq 0 \text{ and } k = 284 \quad (9)$$

Combining the "most recent past" method for the observation with $\overline{X}$ for the expected value, we get the

following formula that calculates the statistics for the Canberra technique:

$$C_n = \sum_{i=1}^{k} \frac{\left| O_{(i,n)} - \overline{X}_i \right|}{\left(O_{(i,n)} + \overline{X}_i \right)} \quad (10)$$

A nice property of this multivariate technique is that for number of variables larger than 30, C_n follows a normal distribution approximately [26]. In our case, number of variables is 284 hence it follows normal distribution without regard of what distribution each of the individual variables follow. We can therefore use the 3-sigma limit to signal whether the observed activity in concern is anomalous or not. The upper limit is calculated using the mean and variance of C_n:

$$Upper\ Limit\ (UL) = \overline{C} + 3\,S_C \quad (11)$$

We apply the following formula for calculating IW value:

$$IW(C_n) = \min\left[1, \frac{C_n}{UL}\right] \quad (12)$$

If, during testing, C_n value for an observation exceeds this upper limit, the IW level is set to 1 and an alert signal is generated, otherwise the IW level is lower than 1 and the observation is considered to be normal. However, we can not safely guarantee that the "normal" events, that do not exceed the upper limit, are normal events in reality.

To get a better understanding of the robustness of Canberra technique, we arranged the testing data in three different ways. In the first method, we put all the training events and then all the testing events - an ideal testing environment for the techniques. In the second method we divide the training events into two halves and put the attack events in between. There are seven attack sessions inside the attack events. The seven attack sessions contain 215, 225, 54, 36, 413, 247 and 35 attack events respectively. We put 400 normal events followed by the first four attack sessions (530 attack events), then the rest of the normal events (303 events) followed by the remaining 3 attack sessions (695 attack events). This way of organizing the events during testing tells us how Canberra technique performs when normal and attack events are intermixed and the length of normal and attack sessions are nearly equal. In the third way of testing we put 100 normal events before each attack session. This way of mixing the normal and attack events shows us how Canberra technique performs in a noisy environment where much of the attacks are concealed inside the normal events.

VI. Results

We now present the results obtained in the three testing cases – ideal, mixed and noisy. We plot Receiver Operating Characteristics (ROC) curve using the IW values. The ROC approach analyzes the tradeoff between false alarm and detection rates for detection systems. It was developed in the field of signal detection [22] [23]. It has now become the standard approach to evaluate detection systems and have been used in language and speaker identification [24] and medical risk prediction [25]. ROC curves for intrusion detection indicate how the attack detection rate changes as internal threshold are varied to generate more or fewer false alarms to tradeoff detection accuracy against analyst workload. Measuring the detection rate alone indicates only the types of attacks that an intrusion detection system may detect, it does not indicate the human workload required to analyze false alarms generated by normal background traffic. Low false alarm rate with high detection rate means that the detection output can be trusted and human labor required to confirm any detection is minimized.

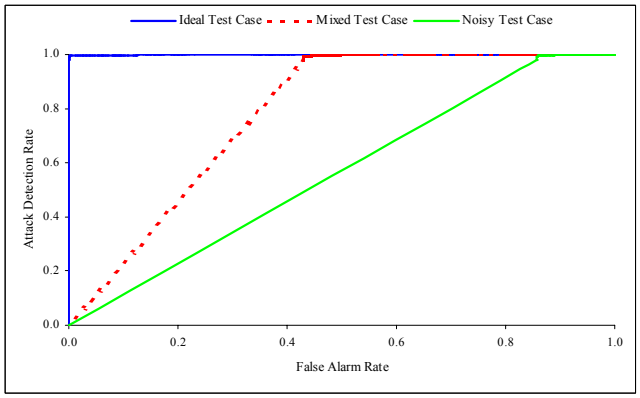

Figure 1 ROC Curves for Canberra Metric in Three Test Cases

From the figure we understand that Canberra technique performs ideally in the ideal test case. It achieved 100% attack detection rate at 0% false alarm rate. Its performance deteriorates in the mixed case and achieves 100% detection rate only after 42% false alarm rate. In the noisy test case it gets to 100% attack detection rate after 85% false alarm rate.

From the result it is evident that Canberra technique perform very good only when normal and attack events are widely separated. When normal events are intermixed with attack sessions it does not perform that well and it misses many attacks and raises too many false alarms. And in the noisy test case, it treats the normal events as noise and as a result false alarm increases. Therefore, we conclude that Canberra technique is good only at special condition and does not perform at acceptable level in all cases.

VII. Future Direction

The data set used in this study corresponds only to 5 minutes of BSM audit data. We need to apply a multi-day data set to test the scalability of the Canberra technique as well. It will be interesting to find out how Canberra technique performs in detecting small attack sessions amidst of huge number of normal events. In our study we maintained a singe event stream for both normal and attack sessions during training and testing. As our next step we will maintain separate event stream for normal and attack sessions during training and

ISBN 0-7803-9814-9/$10.00 © 2001 IEEE

investigate how it affects the performance of the Canberra technique.

VIII. SUMMARY

In this paper we have described the Canberra technique and its implementation and applied it towards detecting attacks in a small subset of the 1998 MIT Lincoln laboratory data. We tested the performance of Canberra technique in three test setup – ideal, mixed and noisy. We presented the performance of the technique in these three cases through ROC curve. We found out that Canberra performs very well only in ideal case, its performance is not acceptable in other cases.

ACKNOWLEDGMENT

This work is sponsored by the Air Force Office of Scientific Research (AFOSR) under grant number F49620-99-1-0014, and the Defense Advanced Research Projects Agency (DARPA) under grant number F30602-99-1-0506. The U.S. government has the authority to reproduce and distribute reprints for governmental purpose notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either express or implied, of AFOSR, DARPA, or the U.S. Government.

REFERENCES

[1] H. Debar, M. Dacier, and A. Wespi. "Towards a taxonomy of intrusion-detection systems," Computer Networks, 31, pp. 805-822, 1999.

[2] T. Escamilla. *Intrusion Detection: Network Security beyond the Firewall.* New York: John Wiley & Sons, 1998.

[3] G. Vigna, S. Eckmann, and R. Kemmerer. "The STAT Tool Suite." In *Proceedings of the DARPA Information Survivability Conference and Exposition.* Los Alamitos, CA: IEE Computer Society, pp. 46-55, January 2000.

[4] S. Kumar. *Classification and Detection of Computer Intrusions.* Ph.D. Dissertation, Department of Computer Science, Purdue University, West lafayette, Indiana, 1995.

[5] W. Lee, S. J. Stolfo, K. Mok. "Mining in a data-flow environment: Experience in network intrusion detection." In *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '99),* San Diego, CA, August, 1999, http://www.cs.columbia.edu/~sal/JAM/PROJECT/.

[6] D. Anderson, T. Frivold, and A. Valdes. *Next-generation Intrusion Detection Expert System (NIDES): A Summary.* Technical Report SRI-CSL-97-07. Menlo Park, CA: SRI International, May, 1995.

[7] P. Neumann, and P. Porras. "Experience with EMERALD to date." In *Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring,* Santa Clara, California, April, 1999, pp. 73-80, http://www.csl.sri.com/neumann/det99.html/.

[8] Anderson, D., T. Frivold, and A. Valdes, 1995. Next-generation Intrusion Detection Expert System (NIDES): A Summary. Technical Report SRI-CSL-95-07, Computer Science Laboratory, SRI International, Menlo Park, California.

[9] Neumann, P., and P. Porras, 1999. Experience with EMERALD to date. Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring, 73–80. Santa Clara: The Advanced Computing Systems Association.

[10] Javitz, H.S., and A. Valdes, 1991. The SRI statistical anomaly detector. Proceedings of the 1991 IEEE Symposium on Research in Security and Privacy. Oakland: IEEE Computer Society Press.

[11] Javitz, H.S., and A. Valdes, 1994. The NIDES Statistical Component Description of Justification. Technical Report A010, Computer Science Laboratory, SRI International, Menlo Park, California.

[12] Jou, Y., F. Gong, C. Sargor, X. Wu, S. Wu, H. Chang, and F. Wang, 2000. Design and implementation of a scalable intrusion detection system for the protection of network infrastructure. Proceedings of the DARPA Information Survivability Conference and Exposition, 69–83. Los Alamitos: IEEE Computer Society Press.

[13] Forrest, S., S.A. Hofmeyr, and A. Somayaji, 1997. Computer Immunology. Communications of the ACM. 40(10): 88–96.

[14] Ko, C., G. Fink, and K. Levitt, 1997. Execution monitoring of security-critical programs in distributed systems: A specification-based approach. Proceedings of the 1997 IEEE Symposium on Security and Privacy, 134–144. Oakland: IEEE Computer Society Press.

[15] Ghosh, A.K., A. Schwatzbard, and M. Shatz, 1999. Learning program behavior profiles for intrusion detection. Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring. Santa Clara: The Advanced Computing Systems Association.

[16] Lippman, R.P., D.J. Fried, I. Graf, J.W. Haines, K.R. Kendall, D. McClung, D. Weber, S.E. Webster, D. Wyschogrod, R.K. Cunningham and M.A. Zissman, 2000. Evaluating Intrusion Detection Systems: The 1998 DARPA Off-line Intrusion Detection Evaluation. *Proceedings of DARPA Information Survivability Conference and Exposition (DISCEX) 2000*. **2**: 12–26. Los Alamitos: IEEE Computer Society Press.

[17] Kendall, K., 1999. *A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems*. M.S. Thesis, MIT Department of Electrical Engineering and Computer Science.

[18] Ye, N., Q. Chen, S.M. Emran, and K. Noh, 2000. Chi-square Statistical Profiling for Anomaly Detection. In the *Proceedings of IEEE Systems, Man and Cybernetics Information Assurance & Security Workshop*. West Point: United States Military Academy.

[19] Ye, N., Q. Chen, S.M. Emran, and S. Vilbert, 2000. Hotelling's T^2 Multivariate Profiling for Anomaly Detection. In the *Proceedings of IEEE Systems, Man and Cybernetics Information Assurance & Security Workshop*. West Point: United States Military Academy.

[20] Johnson, R.A., and D.W. Wichern, 1998. *Applied multivariate Statistical Analysis*. New Jersey: Prentice Hall. pp. 226-235.

[21] Ryan, T.P, 1989. *Statistical Methods for Quality Improvement*. New York: John Wiley & Sons.

[22] Swets, J.A., 1973. The Relative Operating Characteristic in Psychology. *Science*. 182: 990–1000.

[23] Egan, J.P., 1975. *Signal detection theory and ROC-analysis*. New York: Academic Press.

[24] Martin, A., G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, 1998. The DET curve in Assessment of Detection Task Performance. *Proceedings of EuroSpeech '97*. **4**: 1895–1898.

[25] Lippman, R.P., and D.M. Shahian, 1997. Coronary Artery Bypass Risk Prediction Using Neural Networks. *Annals of Thoracic Surgery*. **63**: 1635–1643.

[26] R. A. Johnson, and D. W. Wichern. Applied Multivariate Statistical Analysis. Upper Saddle river, New Jersey: Prentice Hall, 1998.