A Repeated Sampling and Clustering Method for Intrusion Detection

MWITONDI, Kassim and ZARGARI, Shahrzad <http://orcid.org/0000-0001-6511-7646>

Available from Sheffield Hallam University Research Archive (SHURA) at:
http://shura.shu.ac.uk/16537/

This document is the author deposited version. You are advised to consult the publisher's version if you wish to cite from it.

Published version


Copyright and re-use policy

See http://shura.shu.ac.uk/information.html
A Repeated Sampling and Clustering Method for Intrusion Detection

Mwitondi\textsuperscript{1}, K. and Zargari\textsuperscript{2}, S.

Sheffield Hallam University, Department of Computing\textsuperscript{1,2}, Sheffield, United Kingdom

k.mwitondi@shu.ac.uk and s.zargari@shu.ac.uk

Abstract
Various tools, methods and techniques have been developed in recent years to deal with intrusion detection and ensure network security. However, despite all these efforts, gaps remain, apparently due to insufficient data sources on attacks on which to train and test intrusion detection algorithms. We propose a data-flow adaptive method for intrusion detection based on searching through high-dimensional dataset for naturally arising structures. The algorithm is trained on a subset of 82332 observations on 25 numeric variables and one cyber-attack label and tested on another large subset of similar structure. Its novelty derives from iterative estimation of cluster centroids, variability and proportions based on repeated sampling. Data visualisation and numerical results provide a clear separation of a set of variables associated with two types of attacks. We highlight the algorithm’s potential extensions – its allurement to predictive modelling and adaptation to other dimensional-reduction techniques.

Keywords: Clustering, Cross-Validation, Cyber-Security, Data Mining, Intrusion Detection

1 Introduction
Anomaly intrusion detection deals with detecting of unknown attacks in the network traffic therefore, they are difficult to identify without human intervention. Network administrators struggle to keep up with Intrusion Detection System (IDS) alerts, and often manually examine system logs to discover potential attacks. While various tools, methods and techniques have been developed to deal with intrusion detection and ensure network security, gaps remain, apparently due to insufficient data sources on attacks on which to train and test intrusion detection algorithms. In recent years, data mining and machine learning techniques have widely been adopted within the network security community [1] [2] as the Big Data era entails a greater understanding of the underlying rules for intrusion detection. But, as described below, these developments have brought about both opportunities and challenges.

We propose a data-flow adaptive approach to intrusion detection based on high-dimensional cyber-attacks data described in Section 3.1. Our approach derives from the original ideas in [3] and [4] who laid down the general framework for data mining-driven intrusion detection. The former applied association rules and frequent episodes from audit data for feature selection processes and the latter combined association mining with classification. Both were driven by "the degree of confidence" associated with intrusion detection - the main challenge in knowledge extraction from data as noted in [5]. This paper seeks to uncover the general intrusion behaviour and its allurement to predictive modelling. Its main objectives are

1) To perform dimensional reduction of the data attributes through repeated sampling
2) To comparatively assess the emerging naturally arising groupings and
3) To propose a framework for intrusion detection via unsupervised and supervised modelling

The paper is organised as follows. Section 2 examines previous work relating to the overall behaviour of intrusion and normal activities. It is followed, in Section 3, by the methods – outlining the data sources and the adopted methods. Data analyses, results and discussions are in Section 4 and concluding remarks and recommended future directions are in Section 5.

2 Background
Feature extraction for capturing the overall behaviour of intrusions and normal activities has been practiced for years. Auditing programs were utilised by [6] using data mining techniques. However, their findings, widely published as part of the Defence Advanced Research Projects Agency (DARPA) Intrusion Detection Evaluation Program, do not provide direct solutions to intrusion dynamics in time in space. Randomness training and testing data and its impact model accuracy and reliability – is detailed in [7]. Initiatives to develop single, hybrid, and ensemble classifiers are documented in [8] based on 55 related. Computational intelligence methods such
as artificial neural networks, fuzzy systems and swarm intelligence have also been applied in modelling fault tolerance and error resilience [9]. Similar studies were motivated by the need for developing flexible and adaptive security oriented approaches on different platforms [10].

It is imperative to note that identifying the most relevant features in intrusion detection is not confined to algorithmic computing – it depends much on existing expert domain knowledge skills and the way they are combined with automated tools [11]. This dependency is generally attributed to the overall objective of data mining – extraction knowledge from data. Typically, frameworks for fulfilling that relationship are based on pre-defined ontologies with inherently highly dynamic parameters. The dynamics of these parameters are encapsulated within the domains of unsupervised and supervised modelling for which many applications have been developed in recent years [12]. In the next exposition, we outline an integrated strategy to harmonising inherent dynamics in modelling cyber-attacks via multiple samples.

3 Methods
A high-dimensional dataset is used to explore the general behaviours of cyber-attacks. The dataset comprises of potentially highly correlated variables. The methodology fulfils the following key functions - data understanding, cleansing, clustering and interpretation.

3.1 Data Sources
Data came from thousands of raw network packets of the UNSW-NB 1 15 created by the IXIA PerfectStorm tool in the Cyber Range Lab of the Australian Centre for Cyber Security (ACCS). The dataset, created using twelve algorithms [13] [2], through the process at the top of Figure 1 represent a high-dimensional data which we denote by

\[ \Omega = \left[ \xi_{ij} \right], \quad i = 1,2,\ldots, n \text{ and } j = 1,2,\ldots, p \]  

\[ (1) \]

Figure 1: Data cyber-attack data generation (top) and “normal” and “attack” traffic

where \( n \) is the number of observations and \( p \) is the number of variables. The bottom panel exhibits one of the two class labels of the data with ten levels – 9 different levels as attacks and one as “normal”. We select 82,332 observations on 27 variables – two derived from others as shown in Table 1.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>src</td>
<td>Float</td>
<td>Source ordinal function</td>
</tr>
<tr>
<td>dport</td>
<td>Integer</td>
<td>Source to destination packet count</td>
</tr>
<tr>
<td>spt</td>
<td>Integer</td>
<td>Destination to packet count</td>
</tr>
<tr>
<td>sipo</td>
<td>Integer</td>
<td>Source to destination transaction bytes</td>
</tr>
<tr>
<td>dipo</td>
<td>Integer</td>
<td>Destination to source transaction bytes</td>
</tr>
<tr>
<td>src</td>
<td>Integer</td>
<td>Source transaction rate</td>
</tr>
<tr>
<td>dst</td>
<td>Integer</td>
<td>Destination to source transaction rate</td>
</tr>
<tr>
<td>src</td>
<td>Integer</td>
<td>Source request rate</td>
</tr>
<tr>
<td>dst</td>
<td>Integer</td>
<td>Destination request rate</td>
</tr>
<tr>
<td>sipo</td>
<td>Integer</td>
<td>Source transmitted packet size mean</td>
</tr>
<tr>
<td>dipo</td>
<td>Integer</td>
<td>Destination transmitted packet size mean</td>
</tr>
<tr>
<td>sipo</td>
<td>Integer</td>
<td>Source transmitted packet size mean</td>
</tr>
<tr>
<td>dipo</td>
<td>Integer</td>
<td>Destination transmitted packet size mean</td>
</tr>
<tr>
<td>seqpi</td>
<td>Integer</td>
<td>Source transmitted packet size mean</td>
</tr>
<tr>
<td>seqdi</td>
<td>Integer</td>
<td>Destination transmitted packet size mean</td>
</tr>
<tr>
<td>src</td>
<td>Float</td>
<td>Source start time (seconds)</td>
</tr>
<tr>
<td>dst</td>
<td>Float</td>
<td>Destination start time (seconds)</td>
</tr>
<tr>
<td>sipo</td>
<td>Integer</td>
<td>Source TCP window advertisement value</td>
</tr>
<tr>
<td>dipo</td>
<td>Integer</td>
<td>Destination TCP window advertisement value</td>
</tr>
<tr>
<td>seqpi</td>
<td>Integer</td>
<td>Source TCP sequence number</td>
</tr>
<tr>
<td>seqdi</td>
<td>Integer</td>
<td>Destination TCP sequence number</td>
</tr>
<tr>
<td>sipo</td>
<td>Integer</td>
<td>Source transmitted packet size mean</td>
</tr>
<tr>
<td>dipo</td>
<td>Integer</td>
<td>Destination transmitted packet size mean</td>
</tr>
<tr>
<td>seqpi</td>
<td>Integer</td>
<td>Source transmitted packet size mean</td>
</tr>
<tr>
<td>seqdi</td>
<td>Integer</td>
<td>Destination transmitted packet size mean</td>
</tr>
<tr>
<td>msg</td>
<td>Float</td>
<td>TCP connection setup round trip time (SYN+ACK packet)</td>
</tr>
<tr>
<td>smsg</td>
<td>Float</td>
<td>TCP connection setup round trip time between the SYN and the SYN_ACK packets.</td>
</tr>
<tr>
<td>smsg</td>
<td>Float</td>
<td>TCP connection setup round trip time between the SYN_ACK and the ACK packets.</td>
</tr>
<tr>
<td>atttype</td>
<td>Float</td>
<td>Attack type</td>
</tr>
</tbody>
</table>

Table 1: Selected variables

The multivariate dataset forms a good source of training and testing data

\[ X \subset \Omega | X = [x_{i\tau}], i = 1,2,3, ..., n \text{ and } \tau = 1,2,3, ..., m \leq p \]

The foregoing objectives require searching through the entire sub-space \( X \subset \Omega \) for natural groupings of data \( C_{\text{in}} \). If we let \( K \in X \) be the number of groups, each with a notional probability \( \pi_k \) where \( \sum_{k=1}^{K} \pi_k = 1 \), the \( p \)-dimensional probability function \( p(X, \omega_k) \) is fully described by the parameters in \( \omega_k \), making it possible to merge or split clusters in accordance with adopted measures of similarity. Our strategy seeks to determine group proportions formed by \( c_{\text{in}} \in X \) and their membership which we treat as missing data and by repeatedly sampling from \( X \) we create and validate clusters that characterise the overall behaviour of cyber intrusion.

3.2 Modelling Strategy

Determining the optimal number of clusters in high-dimensional data remains a major challenge in many applications – particularly because data attributes may be highly correlated. A common starting point is usually Exploratory Data Analysis (EDA) which serves to describe the overall data behaviour. Subsequently, dimensional reduction techniques may be applied to reduce the number of variables by combining them. Two popular approaches to dimensional reduction are data clustering – in all its variants and principal component analysis (PCA) [14]. Both methods rely on inherent parameters – typically referred to as “bandwidth” - to determine the underlying structures in data. We conceptualise dimensional reduction via kernel density estimation (KDE) of the probability density function [15] to highlight the bandwidth selection issues and associated smoothing techniques [16]. The KDE is defined as

\[ f_h(x_{\text{in}}) = \frac{1}{n} \sum_{i=1}^{n} K_h(x - x_{i\tau}) = \frac{1}{nh} \sum_{i=1}^{n} K \left( \frac{x - x_{i\tau}}{h} \right) \]

where \( \int K_h(x - x_{i\tau}) \, dx = 1 \) and it is centred at 0 and \( h > 0 \) is a smoothing parameter that determines the exact shape of the kernel.

The choice of \( h \), typically, determines the number of modes in the random variable. Its optimal value is therefore proportional to the expected value of the difference between the fitted and actual functions \( E \left[ \left( \hat{f}_h(x) - f_h(x) \right)^2 \right] \).

The KDE is a special case of clustering with the multivariate case, where \( c_{\text{in}} \) combine to form natural groupings in \( X \) and can be compared to the class variable i.e., \( c_{\text{in}} \propto x_{i\tau} \in X \). Thus, if we let \( x_{i\tau} \in C_i = \{1, 2, ..., K\} \) denote group membership, then this membership is associated with an unobservable indicator variable \( z_i \) defined as

\[ z_i = \{0, 1\}^K = (z_{i1}, z_{i2}, ..., z_{iK}) \text{ such that } z_{ik} = \begin{cases} 1 & \text{if } C_i = k \\ 0 & \text{if } C_i \neq k \end{cases} \]

Since the unobserved variable associates each \( x_{i\tau} \) with a data cluster \( C_{\text{in}} \), our strategy will be seeking to optimise the expected value of \( h \) via repeated sampling. Each sample generates a correlation matrix that is re-ordered based on the angular order of the eigenvectors

\[ a_i = \begin{cases} \tan(e_{i2}/e_{11}) & \text{if } e_{11} > 0 \\ \tan(e_{i2}/e_{11}) + \pi & \text{Otherwise} \end{cases} \]

where \( e_1 \) and \( e_2 \) are the largest eigenvalues from the matrix. For each sample we estimate cluster proportions, centres and variability as follows

\[ \hat{\pi}_k = \frac{\sum_{i=1}^{N} z_{ik}}{n} ; \hat{\mu}_k = \frac{\sum_{i=1}^{N} z_{ik} x_{i\tau}}{\sum_{i=1}^{N} z_{ik}} ; \hat{\sigma}_k^2 = \frac{\sum_{i=1}^{N} z_{ik} (x_{i\tau} - \hat{\mu}_k)^2}{\sum_{i=1}^{N} z_{ik}} \]

The updated parameters are then used for intermediate cross-validation of the emerging structures which are subsequently interpreted and labelled. Final validation is done via comparison with known structures of attack types as outlined below.

4 EDA, Dimensional Reduction and Visualisation

The data in Table 1 are labelled with ten different data-flow levels – one normal and nine intrusions. This class label has been set as binary – pooling together all 9 attack types. This section provides detailed graphical and numerical findings generalising \( \Omega \). The ultimate purpose is to identify the type of attacks that are most pronounced. Figure 1 presents the frequency of the nine attack types - Fuzzers, Analysis, Backdoors, DoS Exploits, Generic, Reconnaissance, Shellcode and Worms, alongside the normal traffic. Particularly interesting are the variables which contribute to the highest variation in the data. In predictive terms, these impinge on the nature of the traffic and so they can be used as predictors of the phenomenon that they symbolise. The exhibits the correlation
matrix re-ordered on its angular order of its eigenvectors which exhibits the separation of the variation. These patterns are averaged over repeated runs and the emerging structures are validated via comparison with attack labels.

Figure 2: Original correlation matrix (LHS) and its re-ordered version on the RHS
The emerging structures in Figure 2 provide insights into how different transmission parameters interact based on which we adjust the adopted cut-off point – in our case, the “tol” parameter in the prcomp() in R – for extracting dominant patterns within $X \subset \Omega$. As PCA deals with multicollinearity through its orthogonal properties, repeated sampling is only concerned about variability and so the tuning parameter “tol”, in this case, which is a function of the standard deviation and the first component is crucial. Our repeated samples utilised a range of levels each projecting testing data onto the PCA coordinate basis.

Figure 3: Plot of the variable loadings - rotations
The most challenging part of PCA is interpreting the extracted components. Among the crucial outputs of the prcomp() function in R is the matrix of variable loadings – called rotations whose columns are eigenvectors of the linear transformation with a direction that is invariant to the transformation. Figure 3 presents a visual pattern of the rotations - the higher the absolute values the more important a variable is – with the combinations of positive and negative values being interpreted as “mixed”. These rotations help in interpreting the extracted components with their different proportions of loaded variables. Since the rotations occur in a subspace of the retained components, the rotation results are affected by the choice of the subspace. Hence, to attain a robust interpretation of the rotation, it is strongly recommended to try several sizes for the subspace. It therefore reasonable to consider averages over multiple random runs.

Figure 4: Attack type projections of the highest two components selected from multiple samples
Figure 4 provides a summary of the results from multiple sampling and clustering. These projections of components one and two were selected from repeated random samples. It was generally evident from the repeated samples that there is a great separation of attributes that describe Generic and Exploits attack types. Each group is associated with a probability and so our repeated sampling experimented with a wide range of probabilities - in line with the strategy proposed above for estimating cluster centroids, variability and proportions.

5 Concluding Remarks
We have presented a data-flow adaptive method for intrusion detection based on high-dimensional
dataset. Repeated sampling from a high-dimensional on-line traffic data produced adaptable centroid, variability and proportion parameters. These parameters are generally used to fine-tune potential structures in each sample which are then aggregated over many runs to come up with recognisable components which allure to predictive modelling. Its novelty derives from iterative estimation of cluster centroids, variability and proportions based on repeated sampling from a large cyber-attack data source. Our results show that out of 25 on-line traffic variables, only a handful could be good predictors of attacks. More importantly, of the nine different attack types only two stood out distinctively in relation to the predictors. Without being influenced by the provided levels of attack types, this finding is particularly intriguing for two reasons – it may mean either masking or swamping of attack types.

It is imperative to highlight two important aspects of this study – the mechanics of the algorithm and the data attributes. If we knew the relevant density functions and classes of attack, we would simply observe data flows and make predictions. But in practice we have to estimate these parameters from “random” data and test our algorithms on another random dataset. Our contribution focused on three key parameters – data centroids, variability and proportions. For continuous attributes, these parameters typically derive from the mean and covariance matrices of the normal mixtures while for categorical data they are group proportions. Our choice of the angular order of the eigenvectors was justified by the adopted dimensional-reduction methods used as it would make multiple comparisons to be made using other eigenvalues. There are other ways of re-ordering the correlation matrix to help visualisation of the inherent patterns. This random nature of analytical studies implies that research be a function of sharing data, design, experimental setups and research findings. One way of achieving that goals is to go the way of open source repositories. We expect that this paper will contribute towards that aspiration, attain data enrichment, methods enhancements and open new paths to future studies.

References


