**Online Detection of User’s Anomalous Activities using Logs**

Kusumanjali Somisetty  
 *Data Science and Analystics (MSc)*

*Abstract*—

Securing the Organization’s Confidential Information is always a concern for any Organization. This Paper implements an Machine Learning approach to monitor the Users activities and determine the anomalous Data.The term anomalous data refers to data that are different from what are expected or normally occur. Detecting anomalies is important in most industries. For example, in network security, anomalous packets or requests can be flagged as errors or potential attacks. In customer security, anomalous online behavior can be used to identify fraud. And in manufacturing and the Internet of Things, anomaly detection is useful for identifying machine failures.

Keywords— Anomalous Detection, Machine Learning, PCA, Random Forest

# Introduction

Network security is a foremost issue these days as the network usage is growing in multi-dimensions due to increased use of handheld devices. Intrusion Detection Systems can help detect malign intentions of network users without compromising the security of the host and the network. There are many machine learning algorithms available which can learn from the training data and can generalize when exposed to new untrained data. There are two types of intrusion detection technique, the first one is Misuse Detection that can catch the known attacks and hence works on the offline data and the other is Anomaly Detection which can detect any abnormal behavior and hence can work well on online data. The KDD data set is a standard data set used for the research on intrusion detection systems.

# KDD Dataset

The NSL-KDD data set with 42 attributes is used in this empirical study. This data set is an improvement over KDD’99 data set from which duplicate instances were removed to get rid of biased classification results. This data set has number of versions available, out of which 20% of the training data is used which is identified as KDDTrain+\_20Percent with a total number of 25192 instances. The test data set is identified by the name KDDTest+ and has a total of 22544 instances. Different configurations of this data set are available with variation in number of instances but the number of attributes in each case is 42. The attribute labeled 42 in the data set is the ‘class’ attribute which indicates whether a given instance is a normal connection instance or an attack. Out of these 42 attributes, 41 attributes can be classified into four different classes as discussed below:

* Basic (B) Features are the attributes of individual TCP connections
* Content (C) features are the attributes within a connection suggested by the domain knowledge
* Traffic (T) features are the attributes computed using a two-second time window
* Host (H) features are the attributes designed to assess attacks which last for more than two seconds

# Experimental Setup

***Research Methodology***

The steps followed as part of the research methodology are as follows:

* KDD data set is selected
* Python Sklearn is used to implement the algorithm
* Random Tree is used as a binary classifier for simulation on algorithm classifies the instances as attack or normal

***Used Classifier***

Machine learning is an artificial intelligence technique which consists of a number of algorithms based on which a model can be developed that learns from the input data known as the training data set and helps predict on testing data set. Though there are many classifiers available, but tree based algorithms produce better accuracy in results without requiring much tuning of parameters. In this paper, Random Tree algorithm, a tree based classifier is selected for simulation from past experience. Random Tree is a set (ensemble) of tree predictors that is called forest. This classifier takes the input feature vector, classifies it with every tree in the forest, and outputs the class label that receives the majority of “votes”.

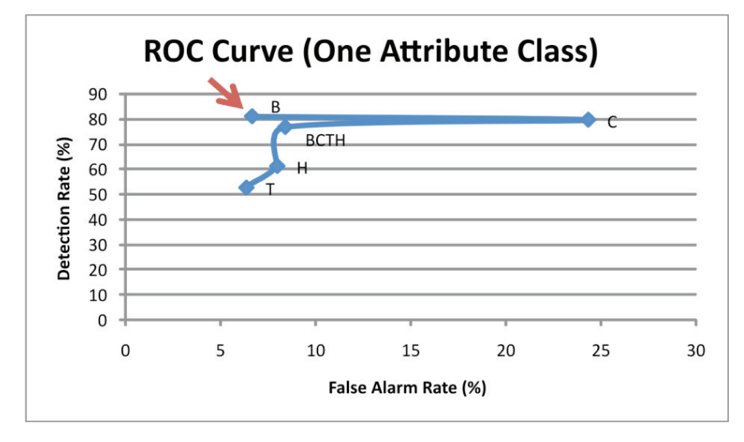
***Metrics***

Intrusion detection metrics helps evaluate the performance of an intrusion detection system21. Some of the commonly used evaluation metrics used with respect to intrusion detection are False Alarm Rate (FAR), Detection Rate (DR), Accuracy, Precision, Specificity, F-score. All these evaluation metrics are basically derived from the four basic attributes of the confusion matrix depicting the actual and predicted classes. These elements of the confusion matrix are:

* True Negative (TN): Number of instances correctly predicted as non-attacks.
* False Negative (FN): Number of instances wrongly predicted as non-attacks.
* False Positive (FP): Number of instances wrongly predicted as attacks.
* True Positive (TP): Number of instances correctly predicted as attacks.

# Simulation Results

In this paper, two of the evaluation metrics that are considered for this study are FAR which is defined as the rate at which normal instances are classified as anomalous and DR which is defined as the ratio of number of instances of correctly predicted attacks to the total number of actual attack instances.



Key observations from ROC curve are as follows:

* Basic class attributes show higher DR
* Traffic attributes show lower FAR
* Content attributes show higher FAR
* Host attributes show low DR but decent FAR

Fig below shows the bar graph presenting DR and FAR for each of fifteen sets of data

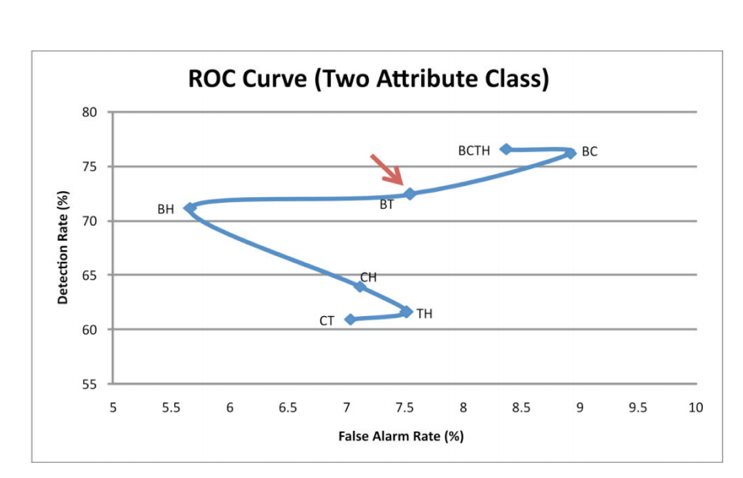
A screenshot of a cell phone

Description automatically generated

***Fig: Graph presenting DR and FAR***

Key observations from ROC curve of Fig. 3 are as follows:

* BC class attributes show DR equivalent to BCTH but has higher FAR
* Without basic class attributes, DR drops



***Fig 3***

##### Acknowledgment

My heartful thanks to professor for Guiding me to do this Project.

##### References

1. Data mining log file streams for the detection of anomalies

<https://pdfs.semanticscholar.org/6f31/428d7b0e9e80680370c4b844cfa2bfb31ba1.pdf>

1. Experience Report: System Log Analysis for Anomaly Detection

http://jmzhu.logpai.com/pub/slhe\_issre2016. K. Elissa, “Title of paper if known,” unpublished.

1. Anomaly Detection Using Persistent Homology

<https://www.computer.org/csdl/proceedings-article/cybersecsym/2016/07942418/12OmNx4yvyt>.

1. Hypergraph-Based Anomaly Detection in Very Large Networks

https://www.computer.org/csdl/proceedingsarticle/cybersecsym/2016/07942418/12OmNx4yvyt

[5] Cyber Anomaly Detection Using Graph-node Role-dynamics

https://arxiv.org/pdf/1812.02848.pdf

**.**