Computers and Electrical Engineering 100 (2022) 107955

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

A Feature Similarity Machine Learning Model for DDoS Attack Detection in Modern Network Environments for Industry 4.0

Swathi Sambangi [a,b,*], Lakshmeeswari Gondi [a], Shadi Aljawarneh [c]

[a] Department of Computer Science and Engineering, GITAM Institute of Technology, GITAM (Deemed to be University), Visakhapatnam, Andhra Pradesh, India
[b] Department of Information Technology, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana, India
[c] Software Engineering Department, Jordan University of Science and Technology, Irbid, Jordan

ARTICLE INFO

Keywords:
Machine learning
Cloud security
DDoS Attack
DoS attack
Industry 4.0
Early detection
Botnet
Feature transformation
Classification

ABSTRACT

Recent advancements in artificial intelligence and machine learning technologies have laid the flagstone for the fourth industrial revolution, Industry 4.0. The industry 4.0 is at a very high momentum when compared to previous revolutions witnessed by humans in a way which was never anticipated. Cyber Physical Systems and Cloud computing are the basis for Industry 4.0. An ongoing research challenge in cloud computing is the immediate need to address security and data availability challenges coined in modern networking environments. For instance, DDoS attacks in cloud are continuously throwing new challenges to network community which makes detection of these attacks, an ongoing research challenge with respect to cloud security. At the outset, the research reported in this work has addressed three important contributions (i) A new gaussian based traffic attribute-pattern similarity function for evolutionary feature clustering to achieve feature transformation-based dimensionality reduction, (ii) A Gaussian based network traffic similarity function for similarity computation between network traffic instances and (iii) A machine learning model SWASTHIKA which uses feature transformation traffic for detection of low rate and high-rate network attacks. For experimental study, the most recent benchmark dataset namely IoT DoS and DDoS attack dataset available at IEEE Dataport is considered as this dataset has highly non-linear traffic instances which are like the real-world traffic. The performance evaluation of the proposed machine learning model SWASTHIKA is done by considering various classifier evaluation parameters such as accuracy, precision, detection rate, and F-Score. The experiment results proved that the attack detection rate of SWASTHIKA is significantly better compared to state of art machine learning classifiers.

1. Introduction

The current advances in information technology, communication technology, artificial intelligence (AI) and machine learning (ML) technologies have laid the flagstone for possibility of the fourth industrial revolution also known as the industry 4.0. AI and ML based systems are wavering the way we interact with computers for information processing. The fusion of artificial intelligence and machine learning based systems is crucial to the development of Industry and can pave a way to new research directions. Hence, there is

* Corresponding author at: Department of Information Technology, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana 500090 India.
E-mail address: ssambangi555@gmail.com (S. Sambangi).

https://doi.org/10.1016/j.compeleceng.2022.107955
Received 25 November 2021; Received in revised form 19 March 2022; Accepted 23 March 2022
Available online 1 April 2022
0045-7906/

emerging necessity to come up with future proof types of artificial and machine learning based designs, architectures, models, services, proof of concept and applications which can secure cyber physical systems and devices.

Industry 4.0 includes Cyber Physical Systems (CPSs), Cloud computing, Internet of things (IoT) and Cognitive computing. Industry 4.0 attributes to smart intelligent networking of processes, machines and essentially aims at improved automation, human-to-machine communication, machine-to-machine communication through design and deployment of smart industrial applications which utilize new age artificial intelligence, machine learning technologies. Industry 4.0 platforms makes it essential for academia, research outputs, industries from diverse domains to work together for design and deployment of modern-day smart applications by fusion of Artificial intelligence, Machine learning, IoT and Data Engineering technologies. For this reason, there is an indispensable need to design, develop and implement proof types of smart applications by fusion of artificial intelligence and machine learning technologies. Thus, one of the crucial and immediate challenges that must be addressed by Industry 4.0 is securing smart networks (for example, smart machines).

The fourth industrial revolution is at a very high momentum when compared to previous revolutions witnessed by humans in a way which was never anticipated. Industry 4.0 has beclouded the thin line between digital world and physical world with the emergence of mezzanine technologies which includes Artificial Intelligence (AI), Cloud Computing, Internet of Things (IoT), advancements in mobile technology, biotechnology, robotics, quantum computing and many more. Cyber Physical Systems (CPSs) is basis for Industry 4.0. With advancements in Industry 4.0, it is estimated by 2025, irrespective of any industry and businesses, there shall soon be a major paradigm shift to cloud-based technology which is a principal and decisive player in the development of Industry 4.0. Cloud computing helps businesses and industries to adapt to new technologies and evolve quickly, thus unlocking and exploring those to technological advancements and opportunities that Industry 4.0 can provide. Hence, the real challenge that must be addressed by Industry 4.0 is the security of Cyber Physical Systems (CPSs) and Cloud environments.

One of the evolving research challenges in cloud computing and its allied research areas such as IoT, SDN and Fog is the immediate need to address (i) network security issues and (ii) network data availability challenges thrown by modern day networking environments. For instance, DDoS attacks in cloud networks are coining new issues and challenges to network community. These issues and challenges make a DDoS attack detection task an important research challenge with respect to cloud security. A primary reason for issues and challenges faced by detection systems is the high non-linearity in the real-world network traffic. A network attack which is instigated by an attacker in Cloud and IoT environments is predominantly targeted to make the available services vulnerable. Two popular network attack types that throw huge challenges to network community in modern environments are the (i) Denial-of-Service attacks (DoS attacks) and (ii) Distributed Denial of Service attacks (DDoS attacks). When a distributed network attack event such as the DDoS attack occurs in a cloud network, the target network service (or network site) affected by the network attack becomes suddenly slow, and sometimes the network site or network services are unavailable until the network site or services are recovered following the network attack. Nevertheless, sometimes it is also noticed that a legitimate increase in the network traffic is also one of the reasons behind internet sources becoming vulnerable. Thus, differentiating and judging between legitimate network traffic and attack traffic is an essential and immediate need.

As per the public report by amazon web services (AWS), the massive DDoS attack that has been recorded to date is during February 2020 [4–6]. In this case, it is noticed that the peak incoming network traffic which is experienced by network because of the DDoS attack is 2.3 Tbps. For accomplishing this task, network attackers utilized hijacked CLDAP Webservers (which are also known as Connection less Lightweight Directory Access Protocol webservers). These webservers are an alternative to LDAP web servers, and it is one of the protocols which is utilized to handle user directories. Another massive DDoS attack that is made prior to this 2.3Tbps attack, was the 1.3 Tbps attack which is second largest DDoS attack. This network attack focused on GitHub by regularly trafficking over and over at the rate of 126.9 million traffic packets per second [6]. This makes the DDoS attack prevention [4], DDoS attack detection [6] and DDoS attack mitigation problems [4–6] as important research problems in reference to cloud computing. Out of these three research problems, the DDoS attack detection problem has received primary significance from academia, research outputs and IT industry.

In the current research literature, there are numerous studies that have focused on various possible approaches, techniques, and methods for detection of DDoS attacks. Despite these studies, the deployment of available methods, techniques and tools could not resist DDoS attacks. Eventually, the cloud environment and its users are targeted. It is now an experienced fact that such attacks are substantially increasing time to time in terms of DDoS attack size and DDoS attack frequencies. If we analyze the basis and reason for these attacks then, one of the most agreed reasons is non-consensus among various end points in a distributed internet network and that a global cooperation cannot be commonly enforced. The second reason is socioeconomic factor. The third reason is that there is no way of ensuring and enforcing a single point deployment that provides the best Defence against the DDoS attacks because of nature of DDoS attacks. All these problems prove the necessity to study and identify reasons behind the failure of available methods [5,6].

Usually, DDoS attacks are attempted by utilizing the advantage of cloud vendor's services such as (a) pay-as-you-go service (b) multi-tenancy service and (c) auto scaling service. As an example, consider a typical cloud network infrastructure in which Virtual machines (VMs) are run in huge numbers by cloud servers to provide un-interrupted cloud services to legitimate cloud users or clients. So, whenever an attack is attempted in cloud, the server can consider such a situation as the event of higher resource utilization. The result is that the respective server attempts to utilize the auto scaling feature in cloud computing. Subsequently, the auto scaling feature offered by cloud could result in huge resource allocation and resource migration so as to address the server overloading problem.

Now, assume an attack scenario wherein the process migration and resource allocation continuously continue as a result of attack. In this case, the cloud attacker eventually turns successful in launching a DDoS attack successfully. These DDoS attacks affects cloud user services either directly or indirectly and eventually implicates financial revenues. Few widely attempted DDoS attacks in cloud

environments are (a) Buffer overflow; (b) IP spoofing; (c) land attack; (d) Ping of death; and (e) SYN flooding. In general, a DDoS attack defense can be accomplished using three approaches in cloud environments. They are (i) Detecting network attacks; (ii) Preventing the network attacks; and (iii) network attacks mitigation. Usually, to detect and prevent DDoS attacks botnet detection and anomaly detection techniques are utilized.

A common misconception related to DDoS attacks is that these attacks target mainly web servers, but it is a fact that DDoS attacks can target any computer system which is connected to the internet. Modern networking environments such as Cloud, IoT, Fog and SDN face various network security challenges due to heterogeneity nature of modern network environments. At an abstract level, DDoS attacks may be viewed as a clogged unexpected traffic [6] on a highway which are fundamentally aimed to prevent the regular traffic arrival at its destination. DDoS attacks are usually performed using a network of connected machines. Thus, a major challenge arising during such attacks are made lies mainly in discriminating between normal traffic and attack traffic. For example, each bot acts as if the bot is a legitimate one. DDoS attacks affect is not only bounded to distributed networks, but these attacks are also extended to Cloud networks and IoT environments which are basis for Industry 4.0. Hence, there is an emerging need to address anomaly and intrusion detection challenges faced by Cloud and IoT environments. Some recent and significant research contributions on cloud security using ML techniques and security in IoT environments is discussed in next subsection.

1.1. Related Works

This sub-section summarizes some important research works that motivated the present work reported in this research. It is an accepted fact that Cloud computing and Software defined network (SDN) paradigms have received a wide research interest and acceptance from IT industry, academia, and network research community. On the other hand, the wider acceptance of Cloud and SDN paradigms is hampered by the growing volume of massive security threats. One of the reasons for this is that recent advancements in currently available processing facilities are implicitly helping network attackers to make successful network attacks in various dimensions. For example, the conventional DoS attacks are now unfolded to cloud environments as DDoS attacks [1]. A recent study by Mahjabin and Xiao et al. [1] described (i) various state-of-art studies on distributed DoS attacks, prevention mechanisms, mitigation mechanisms and (ii) various research directions and open research gaps and issues in identification of DDoS attacks.

With massive security threats that are faced by modern network environments (such as IoT, SDN and Cloud computing), there is an emerging demand and need to come up with AI and machine learning based security solutions that have a better reliability when compared to existing security solutions. This is because majority of existing intrusion and anomaly detection systems are designed by considering popular benchmark datasets that do not meet the assessment and evaluation criterion which should be considered during the design of intrusion and anomaly detection systems [4,5]. The performance of intrusion and anomaly detection systems rely on the dataset criteria taken into consideration during the design and development of detection systems. Thus, reliable network traffic datasets play a decisive role in design of efficient intrusion detection and prevention systems. For instance, though, numerous IDS benchmark datasets such as DARPA98 dataset, KDD99, NSL-KDD 19, NSL-KDD 41, ISC2012 and ADFA2013 are used in experimental studies but none of these benchmark datasets satisfy the evaluation and assessment criterion that must be satisfied to consider these datasets to build efficient IDS systems [2]. Past research on IDS systems have not concentrated on evaluation and assessment of IDS datasets used for experimental study and analysis. This is true even in the case of most of the recent research contributions available in the literature. In this direction, a recent significant contribution related to IDS study is by Amirhossein Gharib et al. [2]. To build efficient Intrusion detection (IDS) and Intrusion prevention systems (IPS), Amirhossein Gharib et al. [2], have identified and listed eleven (11) criterion and framework which are vital for assessment, evaluation and selection of network traffic datasets used for experimental study. In [3], Sharafaldin et al. has considered CICIDS2017 dataset for studying nonlinearity behavior of network traffic data. CICIDS 2017 [3] dataset satisfies 11 assessment criteria and evaluation criteria of network traffic datasets for designing efficient IDS for cloud and IoT environments. Swathi et al. [4] presents a multiple linear regression model by considering CICIDS 2017 Friday traffic log. The regression model performance is studied by considering statistical performance evaluation measures such as R^2, RMSE, MSE, MAE, Residual error. The evolution of DDoS attacks, the need and importance to address techniques for DDoS attack identification is discussed by Swathi et al. [5,6]. Gupta et al. [7] presented an overview of DoS and DDoS attacks, various issues, and challenges faced by cloud networks. Gupta et al. [7] also, described some of the possible defense mechanisms, tools, and devices for addressing these attacks and some open research challenges to defend cloud network environment from network attackers. In research study by Sharafaldin et al. [8], a new dataset namely CICIDS 2019 is contributed by researchers to evaluate IDS systems and algorithms on DDoS attacks. For this, a new DDoS taxonomy is proposed for the application layer. The dataset is generated to overcome limitations, weakness, and shortcomings of existing datasets. The recent research study [9] by Sharafaldin et al. outlines a framework for visualizing network attacks. The contribution by Aljawarneh, S.A et al. [10] has proposed GARUDA, a novel Gaussian dissimilarity function to identify network attacks and a detection approach to detect intrusions in cyber physical systems which are the basis for industry 4.0. The research [10] paves a path for design and development of new distance and similarity functions to come out with computationally efficient network anomaly detection systems which is motivation for the research addressed in the present research. A recent study by Kurniabudi et al. [11], carried a comprehensive analysis on significant and relevant features for anomaly detection. The research by Swathi et al. [4,5] presented a multiple linear regression model analysis by using the benchmark CICIDS2017 dataset. However, relatively very limited research studies are performed on CICIDS 2019 dataset [6].

One of the massive threats to big data centers, Cloud, IoT environments, and Internet is coined by low-rate DDoS attacks [12]. The threat arises because a low-rate DDoS attack is fundamentally different from DDoS attacks. For example, the aim of DDoS attack at large is to block legitimate traffic through limiting the network infrastructure and network resources by flooding with high traffic volumes. In contrast to high-rate attacks, a low-rate variant of DDoS attack has relatively very less traffic flow which makes it possible

to easily elude the detection mechanisms [12]. Thus, the study by Zhijun et al. [12], throws light on the need to study new mechanisms to detect and defend against low-rate DDoS attacks. In general, DDoS attacks can be categorized into two abstract variants (i) flood-based attacks and (ii) shrew-based attacks. Flood based attacks can be further categorized into high rate and low-rate attacks. In case of a high-rate flood attack, the packet transmission rate is usually higher than 1000 bps whereas in case of a low-rate flood attack, the transmission rate of packets is less than 1000 bps [12]. Alternatively, in shrew based low-rate attacks, the attack flows usually accounts between 10% and 20% of usual traffic. The traditional solutions such as IDS and firewalls are unable to effectively detect complex DoS and DDoS attacks. Reliable security solutions can be designed by integration of artificial Intelligence and machine learning designs and techniques [13]. In this study [13], researchers propose to use the three channel visualization traffic images and have applied Resnet deep learning model for performance analysis and study. The dataset used in the study is obtained by considering non-image CICIDS 2019 network traffic dataset. This network traffic image dataset namely IoT DoS and DDoS attack dataset [14] can be used to test the performance of machine learning and deep learning models. DDoS and Software defined networking (SDN) have differential characteristics. For example, on one side of the coin, SDN capabilities makes it feasible to easily detect and react to DDoS attacks, but on the other side, the security of SDN itself is an issue because of several vulnerabilities which exists across various SDN platforms. In similar lines, the study [15] carried on DDoS attacks in SDN and CLOUD coins an interesting discussion on how recent advances in SDN-based cloud can aid to defeat the DDoS attacks in cloud. The study [15] by Yan et al. thus outlined few open research problems in SDN and cloud environments which may be of interest to researcher community. A recent study by Nassif et al. [16] carried a systematic review on machine learning and security in cloud networks by choosing 63 relevant research studies. The focus was on three aspects. They include (i) various cloud security threats, (ii) machine learning methods employed and (iii) the performance of the employed models. The study identified that there are eleven cloud security areas, thirty machine learning methods and thirteen evaluation metrics are usually employed in the related research studies. The findings of the study prove that, most dominant studies are on DDoS and data privacy. One of the recent research focuses is on identifying low-rate DDoS attacks in IoT, cloud and SDN environments and this is addressed by Zhijun et al. [17]. Another alarming concern is that Low-rate DDoS (LDDoS) attacks are not same as the traditional DDoS attacks. LDDoS attacks have relatively very small traffic flow and hence they can easily escape routers and counter defense deployments employed. Agarwal et al. [18] carried a detailed study on various mechanisms for DDoS attack Defense in cloud environment. A mathematical model is proposed [19] for addressing DDoS attack detection in cloud. Yu et al. [19] also presented various challenges in detection of DDoS attacks in cloud and explores the possibility to beat DDoS attacks in cloud. Similarity measures for temporal pattern mining are proposed by Vangipuram et al. [20,21], which are the main motivation for the present research. In [22], Vangipuram et al. has proposed a machine learning based intrusion detection model for IoT. Deep learning facilitates to obtain improved detection accuracies as traditional machine learning models fail to detect accurately and precisely. Evolution of fifth generation networks (5G networks) has paved a way to the development of IoT industry. 5G networks and IoT facilitated interconnection of almost everything. This convenience to connect almost everything throws the cyber security challenge to secure the devices. Determining the best features and feature combinations is very important in discriminating between benign and malicious traffic (or software). A recent work by Lu et al. [24] proposed malware detection framework which is based on deep neural networks called as DLAMD. The various security challenges in Industrial IoT environments are discussed by Basset et al. [25], For this, they have proposed a model based on forensics called Deep-IFS to detect attacks in IIoT environments.

1.2. Problem Formulation

This sub-section summarizes the proposed work from various viewpoints such as the problem definition, research gap and motivation.

1.2.1. Problem Statement

Given, a high dimensionality multi-variate network traffic dataset which represents normal network traffic and attack traffic, the research problem is to study the degree of non-linearity in the traffic data and propose a computationally efficient machine learning model which employs feature reduction and that it can detect network attacks in incoming network traffic with better accuracy, precision, and detection rates compared to state-of-art machine learning based detection models.

1.2.2. Research Gap and Motivation

Though, several research contributions have addressed identification and detection of network traffic attacks in Cloud, IoT, and other modern network environments, yet there are several research gaps. The various research gaps that are identified during the study are stated below.

(a) Currently, research contributions related to network anomaly detection and intrusion detection, straight away utilize network traffic datasets. A detailed study on nature of datasets such as non-linearity of traffic is not attempted. A detailed study on datasets can give some key insights and possibility to build and design better detection models.
(b) Current research studies that have addressed machine learning based traffic attack detection in Cloud network environments did not consider the network traffic distribution in designing detection models. There is hence a good scope for promising research in this direction.
(c) Current studies straight away apply distance function (such as Euclidean distance, Cosine) even for high dimensionality traffic data. But these distance functions are unsuitable to gauge high dimensional data [6,10]. Eventually, the detection models so built using these distance functions fail to yield promising results.

(d) Network traffic generated in modern day network environments has high non-linearity. Hence, it is possible for a substantial overlap of normal and attack traffic [13]. Thus, accurate and precise identification of network attack traffic has been one of the highly challenging tasks.
(e) The potentiality of reducing dimensionality of the network traffic through feature transformation, yet maintaining the actual network traffic distribution is not addressed in majority research studies. In fact, the existing studies are limited to feature selection rather than feature reduction.
(f) The true potentiality of regression-based attack detection is underutilized in current research contributions. There is a scope to study the performance of regression-based detection models on dimensionality reduced traffic [4] [5].

All the above-mentioned research gaps in the state-of-art research contributions pave motivation for research contribution presented in this paper. Some important research works that motivated this research are the contributions by Vangipuram et al. [20–22], Shadi et al. [10]. The rest of the paper is outlined as follows. Section 2 presents SWASTHIKA, a machine learning model for low rate and high-rate attack detection; Section 3 summarizes the need to understand network traffic data non-linearity degree by plotting Andrews's curves for various benchmark datasets; Section 4 presents the experiment results and discussions. Finally, the section 5 concludes the research contribution of this paper.

2. SWASTHIKA –A Machine Learning Model for Low Rate and High-Rate DDoS Attack Detection

The proposed machine learning approach SWASTHIKA for network traffic attack detection in modern networking environments such as Cloud and IoT is presented in this section. At section 2.1, a novel similarity function SWATHI is presented to group network traffic feature patterns to achieve traffic feature transformation. This is followed by the machine learning algorithm employed by the proposed ML model at section 2.2 and the framework for network attack detection at section 2.3. The proposed ML model applies the proposed traffic similarity function to determine similarity between any two network traffic instances to carry attack classification and detection.

2.1. Similarity Function for Feature Clustering of Traffic Attribute Patterns

One of the implicit challenges for machine learning and deep learning algorithms is handling high dimensionality in multivariate vector instances. Dimensionality of multivariate vectors is an important issue which cannot be ignored as they introduce high non-linearity. An important operation in building any machine learning (or a deep learning) model is the similarity computation and hence the task of similarity computation between any two given multi-dimensional vector instances lies at the heart of machine learning (or deep learning) algorithms. Thus, application of an appropriate similarity function could be a game changer in machine learning and deep learning algorithms. For example, choosing a better similarity function can reduce overall false positives and false negatives of the learning model. The reduction in number of false positives and false negatives can in turn help us to build an efficient learning model which has better accuracy and precision rates.

Consider the task of detecting the low-rate DDoS attack instances. The low-rate DDoS attack instances throw huge challenges to the learning model in that they show near similarity to normal traffic instances. The near similarity of low-rate DDoS traffic instances makes it very difficult for the learning model to identify the traffic instance as an attack. This problem does not minimize even for high-rate DDoS attacks as the high dimensionality of vector instances is always a primary concern. Feature transformation is one of the machine learning techniques that can help us in achieving an optimal representation of the input dataset. A proper feature representation of the input dataset when fed as input to learning models can help us in designing and building learning models that can yield higher accuracy and precision. Similarity computation can play a vital role in achieving optimal representation of the dataset.

Thus, to carry accurate and precise similarity computations, an appropriate similarity function is necessary. For this, a novel traffic attribute similarity function is introduced at subsection 2.1.2. At subsection 2.1.3, a similarity function is proposed to obtain the similarity between network traffic instances. The following subsection 2.1.1 outlines framework for achieving traffic feature transformation.

2.1.1. Traffic feature transformation

Let T_i be an i^{th} network traffic instance and C_i is the known class label of the i^{th} network traffic instance. It is understood that any traffic instance T_i is fundamentally defined using 'm' attributes. Let DS denote a dataset consisting of 'n' labelled traffic instances. To represent an i^{th} network traffic instance with a class label C_i we use the representation given by Eq. (1).

$$T_i = \langle (t_{i1},\ t_{i2},\ t_{i3},\ \ldots\ldots\ldots\ldots t_{im}\ ),\ C_i \rangle \tag{1}$$

Let

$$[T] = \begin{bmatrix} t_{11}\ t_{12}\ t_{13}\ \cdots\cdots t_{1m} \\ t_{21}\ t_{22}\ t_{23}\ \cdots\cdots t_{2m} \\ \cdots\ \cdots\ \cdots\ \cdots\ \cdots\ .. \\ t_{n1}\ t_{n2}\ t_{n3}\ \cdots\cdots t_{nm} \end{bmatrix}$$

and

$$[C] = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ \vdots \\ c_n \end{bmatrix}$$

be the generalized representations of traffic matrix and class label matrix. The labelled traffic dataset can thus be represented as an augmented matrix DS = [T: C].

Thus, we have the generalized representation of network traffic instances dataset as a multidimensional matrix of order equal to n * (m+1) as represented using Eq. (2). Usually, the number of classes in a dataset is less than number of instances as more than one traffic instance can belong to the same class. Here, we assume that there are 'q' distinct class labels.

$$DS = [T : C] = \begin{bmatrix} t_{11} & t_{12} & t_{13} & \cdots & \cdots & t_{1m} & c_1 \\ t_{21} & t_{22} & t_{23} & \cdots & \cdots & t_{2m} & c_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\ t_{n1} & t_{n2} & t_{n3} & \cdots & \cdots & t_{nm} & c_n \end{bmatrix} \tag{2}$$

We can relate the chance of j^{th} network traffic attribute N_{aj} to influence the traffic class C_q by finding the probability denoted by $\Pr\left(\frac{N_{aj}}{C_q}\right)$.

In general, we can represent every network traffic attribute N_{aj}as a 'q' dimensionality vector using Eq. (3). As $\Pr\left(\frac{N_{aj}}{C_q}\right)$ represents the probability value, it can take values between 0 and 1 inclusive. Here, the notation $X_{(N_{aj})}$ represents a traffic attribute pattern.

$$X_{(N_{aj})} = \left(\Pr\left(\frac{N_{aj}}{C_1}\right), \Pr\left(\frac{N_{aj}}{C_2}\right), \Pr\left(\frac{N_{aj}}{C_3}\right), \ldots\ldots\ldots, \Pr\left(\frac{N_{aj}}{C_q}\right)\right) \tag{3}$$

Hence, for a network traffic dataset DS where every traffic instance is defined over 'm' network attributes, we get 'm' traffic attribute patterns. All such traffic patterns may be denoted as a traffic attribute pattern matrix with 'm' rows and 'q' columns of the order m * q. Eq. (4) depicts the traffic pattern matrix obtained for 'm' traffic attributes by considering 'q' classes.

$$X_{(N_a)} = \begin{bmatrix} \Pr\left(\frac{N_{a1}}{C_1}\right) & \Pr\left(\frac{N_a}{C_2}\right) & \Pr\left(\frac{N_a}{C_a}\right) & \cdots & \cdots & \Pr\left(\frac{N_a}{C_q}\right) \\ \Pr\left(\frac{N_{az}}{C_1}\right) & \Pr\left(\frac{N_{az}}{C_2}\right) & \Pr\left(\frac{N_{az}}{C_a}\right) & \cdots & \cdots & \Pr\left(\frac{N_{az}}{C_q}\right) \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \Pr\left(\frac{N_{am}}{C_1}\right) & \Pr\left(\frac{N_{am}}{C_2}\right) & \Pr\left(\frac{N_{am}}{C_z}\right) & \cdots & \cdots & \Pr\left(\frac{N_{am}}{C_q}\right) \end{bmatrix} \tag{4}$$

The traffic attribute pattern data which is represented as a matrix can be clustered to obtain the traffic pattern clusters by using an appropriate traffic attribute pattern similarity function. For example, suppose that the clustering task results in 'z' clusters. Then, we can represent the relation between the 'z' clusters and 'm' traffic patterns as a multi-dimensional matrix of the order m*z as shown below using Eq. (5). In Eq. (5), the notation $T(X_{(N_a)})$ denotes the transformation matrix.

$$T(X_{(N_a)}) = \begin{bmatrix} \mathrm{sim}(X_{(N_{a1})}, G_1) & \mathrm{sim}(X_{(N_{a1}}, G_2) & \cdots & \mathrm{sim}(X_{(N_{a1})}, G_2) \\ \mathrm{sim}(X_{(N_{a2})}, G_1) & \mathrm{sim}(X_{(N_{a2})}, G_2) & \ldots & \mathrm{sim}(X_{(N_{a2})}, G_2) \\ \cdots & \cdots & \cdots & \\ \mathrm{sim}(X_{(N_{am})}, G_1) & \mathrm{sim}(X_{(N_{am})}, G_2) & \ldots & \mathrm{sim}(X_{(N_{am})}, G_z) \end{bmatrix} \tag{5}$$

The transformation matrix $T(X_{(N_a)})$ denoted using Eq. (5) can be used to obtain a new representation of traffic matrix by multiplying the actual traffic matrix [T]with transformation matrix $T(X_{(N_a)})$. The new traffic data matrix can be denoted using Eq. (6).

$$[W] = [T] * T(X_{(N_a)}) \tag{6}$$

In the proposed method, the new traffic data representation denoted using [W] is used to perform classification and prediction by suitably training the learning model.

2.1.2. Proposed Traffic Attribute Similarity Function - Swathi

We now introduce the network traffic attribute similarity function Swathi, which can used to find the similarity between any two traffic attribute patterns. The proposed traffic attribute similarity function is given by Eq. (7).

$$\mathscr{S}wathi\left(X_{(N_{ai})}, X_{(N_{aj})}\right) = \frac{\mathscr{U}_{mean} + \delta}{1 + \delta} \tag{7}$$

Where

$$\mathscr{U}_{mean} = \frac{\sum_{k=1}^{k=m} \mathscr{S}^{k}\left(X_{(N_{ai})}, X_{(N_{aj})}\right)}{\sum_{k=1}^{k=m} \mathscr{T}^{k}\left(X_{(N_{ai})}, X_{(N_{aj})}\right)} \tag{8}$$

and

$$\delta^{k}\left(X_{(N_{ai})}, X_{(N_{aj})}\right) = \begin{cases} \left(1 - \exp\left[\frac{1 - \left(\Pr\left(\frac{N_{ai}}{C_q}\right) - \Pr\left(\frac{N_{ai}}{C_q}\right)\right)}{\sigma_g}\right]^2\right) & ; \Pr\left(\frac{N_{ai}}{C_q}\right) \neq 0 \text{and} \Pr\left(\frac{N_{aj}}{C_q}\right) \neq 0 \\ -\left(1 - \exp\left[\frac{\left[\Pr\left(\frac{N_{ai}}{C_q}\right) - \Pr\left(\frac{N_{aj}}{C_q}\right)\right]^2}{\sigma_g}\right]\right) & ; \Pr\left(\frac{N_{ai}}{C_q}\right) = 0 \text{and} \Pr\left(\frac{N_{aj}}{C_q}\right) \neq 0 \\ -\left(1 - \exp\left[-\left[\frac{\left[\Pr\left(\frac{N_{ai}}{C_q}\right) - \Pr\left(\frac{N_{a}}{C_q}\right)\right]^2}{\sigma_g}\right]\right)\right) & ; \Pr\left(\frac{N_{ai}}{C_q}\right) \neq 0 \text{and} \Pr\left(\frac{N_{aj}}{C_q}\right) = 0 \\ 0 & ; \Pr\left(\frac{N_{ai}}{C_q}\right) = 0 \text{and} \Pr\left(\frac{N_{af}}{C_q}\right) = 0 \end{cases} \tag{9}$$

$$\mathscr{T}^{k}\left(X_{(N_{ai})}, X_{(N_{aj})}\right) = \begin{cases} 0; \ \Pr\left(\frac{N_{ai}}{C_q}\right) = 0 \ and \ \Pr\left(\frac{N_{aj}}{C_q}\right) = 0 \\ 1; else \end{cases} \tag{10}$$

In Eq. (7), the parameter $\mathscr{U}_{mean}$ represents the normalized similarity value among the traffic attribute pattern vectors $X_{(N_{ai})}$ and $X_{(N_{aj})}$. Solving analytically, we have the value of δ equal to 0.6321. In Eq. (9), $\mathscr{S}^{k}(X_{(N_{ai})}, X_{(N_{aj})})$ represents the membership similarity between the k^{th} element of the m-dimensional traffic attribute pattern vectors $X_{(N_{ai})}$ and $X_{(N_{aj})}$.

The notation $\Pr\left(\frac{N_{ai}}{C_q}\right)$ denotes the likely chance of attribute N_{ai} to belong to class C_q. In Eq. (10), $\mathscr{T}^{k}\left(X_{(N_{ai})}, X_{(N_{aj})}\right)$ refers to whether the k^{th} element of the m-dimensional traffic attribute pattern vectors $X_{(N_{ai})}$ and $X_{(N_{aj})}$ is considered to determine the respective membership between traffic attribute pattern vectors $X_{(N_{ai})}$ and $X_{(N_{aj})}$

2.1.3. Network Traffic Similarity Function

We now introduce the traffic similarity function to find the similarity between any two traffic instances. The traffic similarity function is given by Eq. (11).

$$\mathscr{S}imTap\left(T_{(i)}, T_{(j)}\right) = \frac{\mathscr{U}_{mean} + \delta}{1 + \delta} \tag{11}$$

Where

$$\mathscr{U}_{mean} = \frac{\sum_{k=1}^{k=m} \mathscr{S}^{k}\left(T_{(i)}, T_{(j)}\right)}{\sum_{k=1}^{k=m} \mathscr{T}^{k}\left(T_{(i)}, T_{(j)}\right)} \tag{12}$$

and

$$\mathscr{S}^{k}\left(T_{(i)}, T_{(j)}\right) = \begin{cases} \exp^{-\left[\frac{T_{(ik)} - T_{(jk)}}{\sigma_g}\right]^2}; \ T_{(ik)} \neq 0 \ and T_{(jk)} \neq 0 \\ -1; \ T_{(ik)} = 0 \ and T_{(jk)} \neq 0 \\ -1; \ T_{(ik)} \neq 0 \ and T_{(jk)} = 0 \\ 0; \ T_{(ik)} = 0 \ and T_{(jk)} = 0 \end{cases} \tag{13}$$

$$\mathscr{T}^{k}\left(T_{(i)}, T_{(j)}\right) = \begin{cases} 0; \ T_{(ik)} = 0 \ and T_{(jk)} = 0 \\ 1; else \end{cases} \tag{14}$$

In Eq. (12), the parameter $\mathscr{U}_{mean}$ represents the normalized similarity value among the traffic instances $T_{(i)}$and $T_{(j)}$. Solving

analytically, we have the value of δ equal to 1. In Eq. (13), $\mathscr{S}^{k}(T_{(i)}, T_{(j)})$ represents the membership similarity between the k^{th} element of the m-dimensional traffic vectors $T_{(i)}$and $T_{(j)}$.

In Eq. (14), $\mathscr{T}^{k}(T_{(i)}, T_{(j)})$ refers to whether the k^{th} element of the m-dimensional traffic vector $T_{(i)}$and $T_{(j)}$ is considered to determine the respective membership between traffic vectors $T_{(i)}$and $T_{(j)}$.

2.1.4. Working Example

Suppose that we have traffic patterns obtained as $X_{(N_{a1})} = (0.8,\ 0.6,\ 1.0)$ and $X_{(N_{a2})} = (0.3,\ 0.5,\ 0.0)$. Then, we have $\sum_{k=1}^{k=3}\mathscr{S}^{k}(X_{(N_{a1})}, X_{(N_{a2})}) = -0.31035$ and $\sum_{k=1}^{k=3}\mathscr{T}^{k}(X_{(N_{a1})}, X_{(N_{a2})}) = 3$. In this case, $\mathscr{U}_{mean} = -0.10345$. So, the similarity between these two traffic attribute patterns is obtained by applying Eq. (7) as $\mathscr{S}wathi(T_{(i)}, T_{(j)}) = 0.3239$.

2.2. Proposed Machine Learning Algorithm – SWASTHIKA

In this section, the algorithm for feature transformation is outlined. The proposed method applies the traffic similarity functions outlined at Section 2.1. The learning model which uses these similarity functions is named as SWASTHIKA.

Algorithm: Network Traffic Feature Transformation

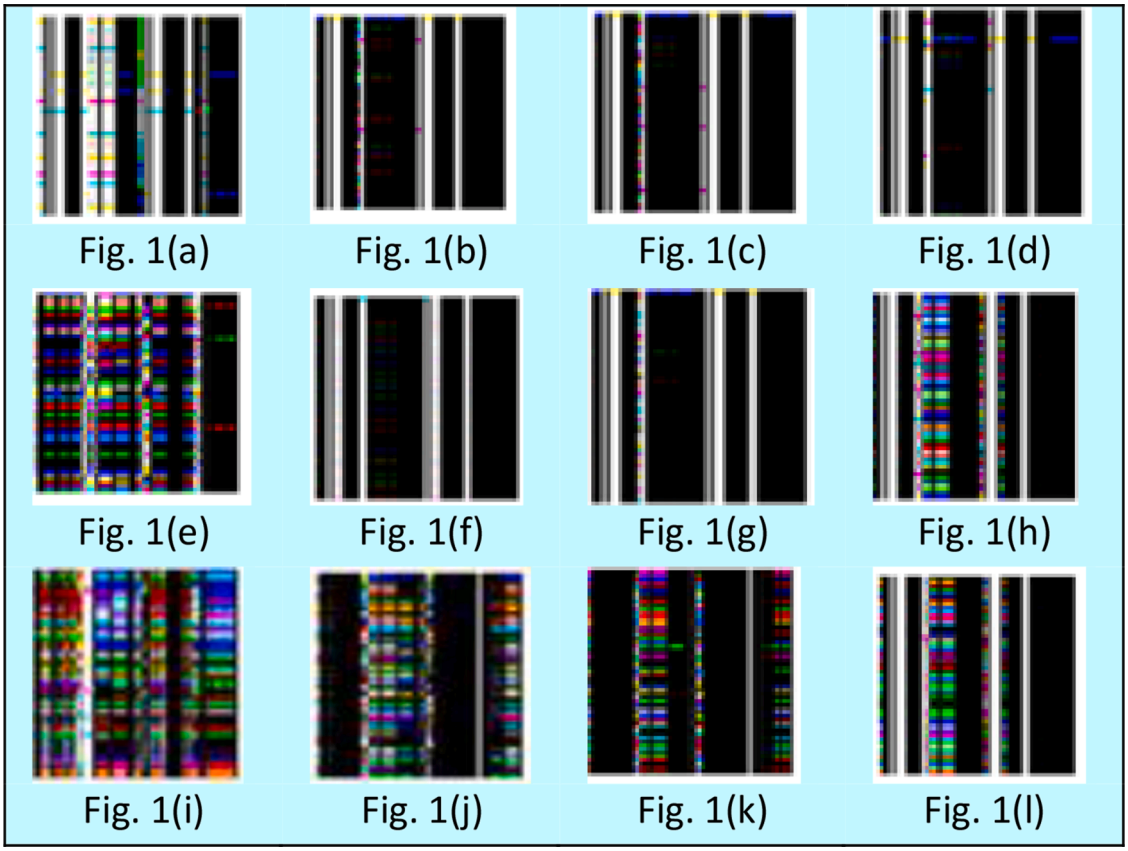
```
Procedure Feature Transformation (augmented traffic matrix, threshold, deviation):
Input:   Training dataset, D consisting of 'N' traffic instances with 'm' dimensions
Hyperparameter:   threshold, deviation
1: Begin
2: for every traffic attribute TA_i
3:    Compute the probability for attribute TA_i to belong to each traffic class C_k
4:       Represent the probability values obtained as traffic attribute pattern vector TP_i
5: End_For
6: for every traffic attribute pattern vector TP_i
7:       Perform evolutionary clustering of traffic attribute patterns in incremental manner
8:       for chosen hyperparameter settings
9:       Store traffic attribute pattern clusters
10: End_For
11: for every traffic pattern cluster G_i
12:       for every traffic attribute TAi
13:          Compute similarity of traffic attribute pattern TP_i to each cluster G_i
14:       End_For
15: End_For
16: Store similarity values of traffic attribute pattern to clusters as a 2D transformation matrix
17: Output the traffic attribute transformation matrix, TM|TA_i |x |G_i|
18: Transformation Traffic Data = [D] NXm * [TM]|TAi |x |Gi|
19: End
20: End Procedure
```

Algorithm: SWASTHIKA (labelled network traffic instances, incoming network traffic)

```
1: Begin
2: Build machine learning model by calling procedure Network Traffic Feature Transformation ()
3: For every new incoming traffic instance
4: Call Procedure Feature Transformation () to feature transform incoming traffic
5: Compute gaussian similarity of incoming traffic to each traffic instance in TTD
6: If (gaussian similarity value >= threshold)
7:       Assign class label to incoming traffic based on computed similarity
8: End_If
9: Allow the incoming traffic if it is classified as a normal traffic
10: Block if it is predicted as attack
11: End_for
12: End
```

2.3. Proposed Model for Network Attack Detection in Cloud and IoT Environments - Swasthika

Fig. 1(a) to Fig. 1(l) depicts sample network traffic images of an IoT DoS DDoS attack dataset [13] which are used for experimental analysis in this research. The traffic classes in the dataset are (a): DNS attack, (b): LDAP attack, (c): MySQL attack, (d): NETBIOS attack, (e): NTP attack, (f): SNMP attack, (g): SSDP attack, (h): UDP Flood attack, (i): Normal Traffic, (j): SYN attack, (k): TFTP attack, (l): UDP Lag. The proposed method for detection of DDoS attack detection in Cloud and IoT environments and any network in general is depicted in Fig. 1(m).



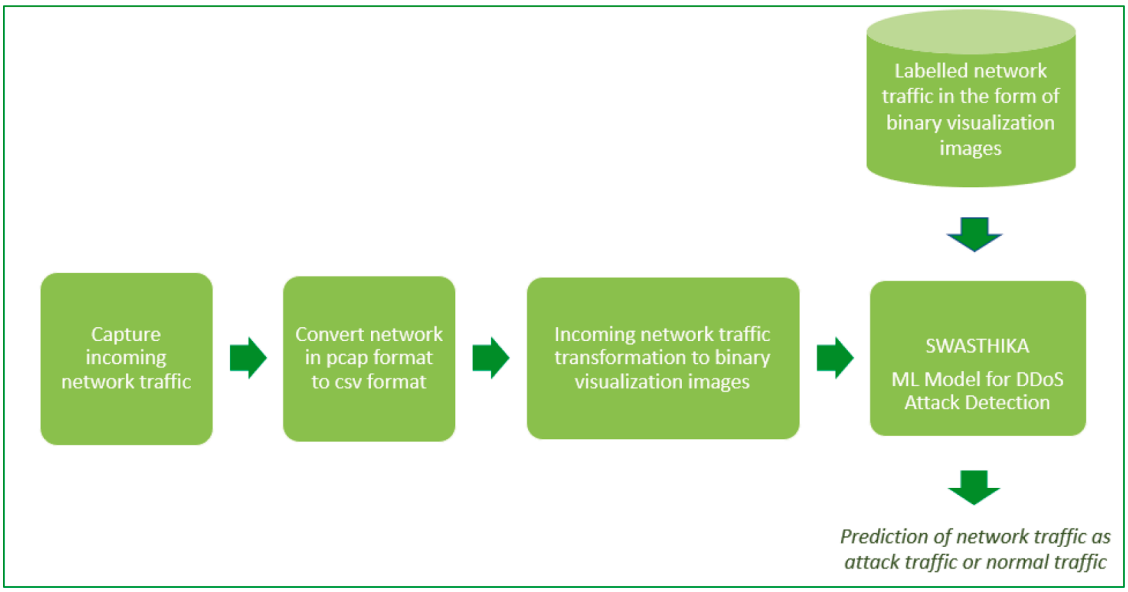


Fig. 1. Images obtained during Data Conversion Stage. (a): DNS attack, (b): LDAP attack, (c): MySQL attack, (d): NETBIOS attack, (e): NTP attack, (f): SNMP attack, (g): SSDP attack, (h): UDP Flood attack, (i): Normal Traffic, (j): SYN attack, (k): TFTP attack, (l): UDP Lag (m) Proposed ML Model for network attack detection in Cloud and IoT

3. Non-linearity in Benchmark Network Traffic Datasets and their complexity

When we build machine learning (or deep learning) based detection models an important concern is to properly analyze and understand the network traffic data utilized to build those detection models [6]. Analyzing the traffic feature complexity of the

Table 1
Function values of three representative functions f1(t), f2(t) and f3(t)

T	-180	-135	-90	-45	0	45	90	135	180
f1(t)	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
f2(t) = sin(t)	0.8	-0.1	-0.9	-0.9	0.0	0.9	0.9	0.1	-0.8
f3(t) = cos(t)	-0.6	-1.0	-0.4	0.5	1.0	0.5	-0.4	-1.0	-0.6

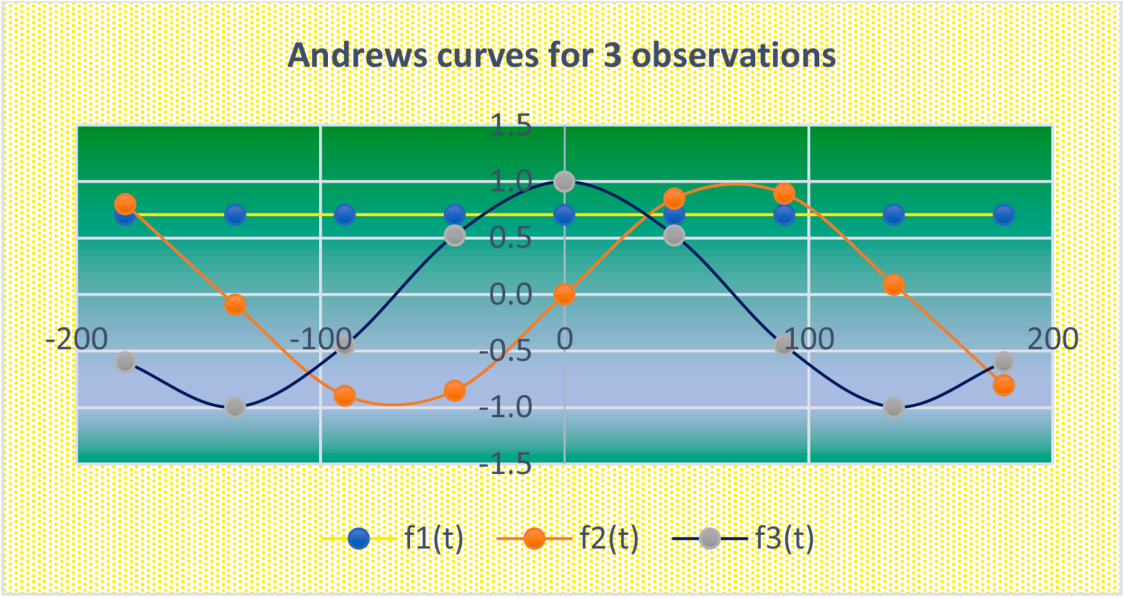


Fig. 2. Andrew curves for three observations using f1(t), f2(t) and f3(t)

network traffic well before aids to design and build better detection models. It is a well-known fact that real world network traffic suffers due to high dimensionality network traffic. For instance, it is quite simple to visualize and understand data with two or three dimensions, but the visualization complexity increases with increasing data dimensionality. A popular and powerful technique to understand high dimensional datasets is to visualize data by plotting Andrew curve [6]. Andrews curves plot is based on Fourier series. An important advantage of Andrew curve plot is that it helps to understand data distribution in high dimension multivariate datasets. The idea behind Andrew curve plot is to represent multivariate data in visual form so that the resulting visual representation helps to understand the dataset. As an example, consider three observations denoted using x1, x2 and x3 as x1 = (1, 0, 0); x2 = (0, 1, 0) and x3 = (0, 0, 1). Now, the idea is to associate a function for each of these three observations. So, we get three functions namely f1(t), f2(t) and f3(t) for observations x1, x2 and x3 respectively. Table. 1 shows function values for f1(t), f2(t) and f3(t) that are obtained by varying 't' from -180 to +180 insteps of 45.

The Andrews curves plotted by using functions f1(t), f2(t) and f3(t) depicted in Table. 1 are represented using Fig. 2. It is now easy to figure out from the visualization depicted in Fig. 2 that observations x1, x2 and x3 are all distinct and are no way similar. Furthermore, Fig. 2 indicates that the first observation (x1) is linear in nature whereas the second (x2) and third observation (x3) are non-linear. In general, visualization of a dataset by plotting Andrew curves can help us to figure out whether the dataset is linear, non-linear, or highly non-linear. This ultimately helps to design better machine learning models.

3.1. Study of Non-linearity in Benchmark Network Traffic Datasets w.r.t Andrews Curve Plot

Visual analysis of benchmark datasets through plotting Andrews plot can help us to know the hidden degree of non-linearity in the benchmark datasets. This can in turn help us to study and design better learning models. Subsections 3.1.1 to 3.1.4 shows the Andrew curve plots for (i) KDD and NSL-KDD, ii) IoT DDoS, iii) SDN DDoS and iv) CICIDS 2017 datasets.

3.1.1. KDD and NSL-KDD datasets

KDD [10] and NSL-KDD [10] datasets are the two popular benchmark datasets which are mainly considered in several research studies to study the performance of learning models which are provided from the University of New Brunswick for the KDD competition. The objective of the competition is to detect bad network connections from good or normal internet traffic.

For this sake, a massive amount of network traffic is captured and bundled into a dataset called KDD dataset. The large number of traffic instances present in KDD dataset makes it possible to study the performance of existing machine learning models and to study

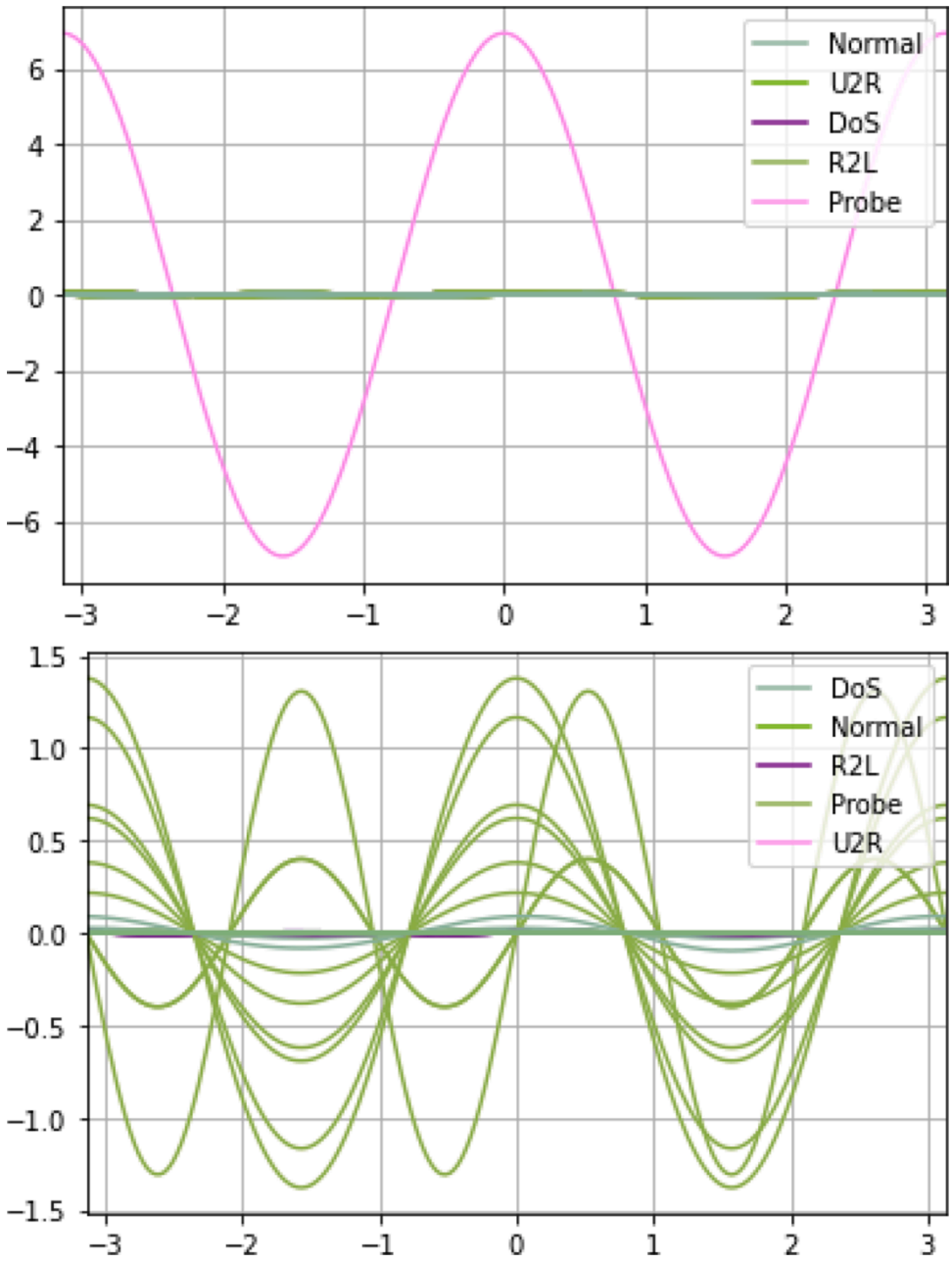


Fig. 3. (a). Andrews curve for KDD-41 Dataset (b). Andrews curve for NSL KDD-41 Dataset (c). Andrews curve for NSL KDD-19 Dataset (d). Andrews curve for KDD-19 Dataset

the performance of future learning models. A revised version of KDD is the NSL-KDD dataset. NSL-KDD is a modern-day internet traffic benchmark dataset which is a revised and cleaned version of KDD dataset. Fig. 3(a) and Fig. 3(b) shows the Andrews curve plot for KDD and NSL-KDD datasets with 41 attributes. Fig. 3(c) and Fig. 3(d) shows the Andrews curve plot for NSL-KDD and KDD dataset with 19 attributes. It is clearly distinguishable that Andrews curve plot of these datasets is moderately non-linear. The non-linearity of datasets throws challenge for learning models to differentiate between normal and traffic instances.

3.1.2. IoT DoS and DDoS Attack dataset – Image representation of CICIDS 2019 dataset

In this subsection, we consider the IoT DoS and DDoS attack dataset available from the IEEE data port [13]. The dataset consisted of 12 traffic classes out of which one traffic class is a normal traffic class. The nature of the IoT DoS DDoS dataset is studied by plotting the Andrews curve plot. Fig. 4 shows the Andrew curves plot which depicts very high non-linearity of the network traffic feature space for

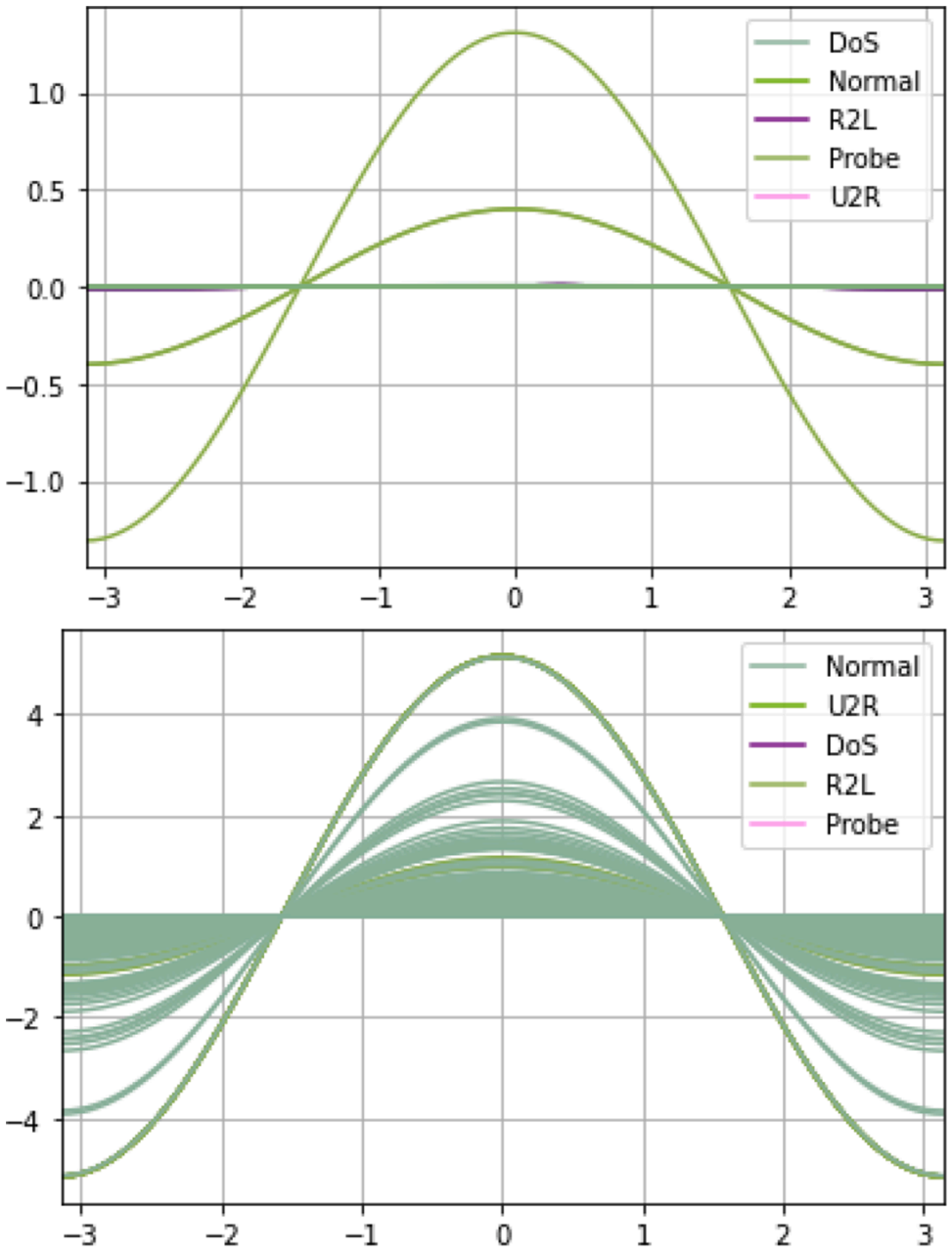


Fig. 3. (*continued*).

IoT DoS DDoS attack dataset wherein each network traffic packet is represented as an image. These image equivalents for each network traffic packet in the IoT DoS DDoS dataset are generated using traffic instances from CICIDS2019 dataset to study the performance of deep learning models [13,14]. It can be clearly understood from the visual representation of the plot shown in Fig. 4, that the network traffic in the IoT DoS and DDoS attack dataset is highly non-linear when compared to KDD and NSL-KDD datasets. The high dimensionality and high non-linearity nature of this dataset makes it a challenging task for the prediction models to differentiate between normal and attack traffic. One of the solutions to reduce high non-linearity nature that exists in the IoT DoS and DDoS dataset is to reduce the dimensionality such that the resulting dataset representation makes it feasible to achieve better performance. For sake of clarity, Fig. 5 shows the Andrews curve plot for IoT DoS and DDoS dataset when the dimensionality is reduced to six (6) dimensions. For these six dimensions, the Andrews curve plot is obtained.

Fig. 5 shows the Andrew curve visualization for reduced dimensions of IoT DoS DDoS dataset which is obtained by using proposed method. From the visualization shown in Fig. 5, it is clearly visible that attack instances in dimensionality reduced data are less non-linear and can be differentiated better when compared to Andrew curve plot shown in Fig. 4.

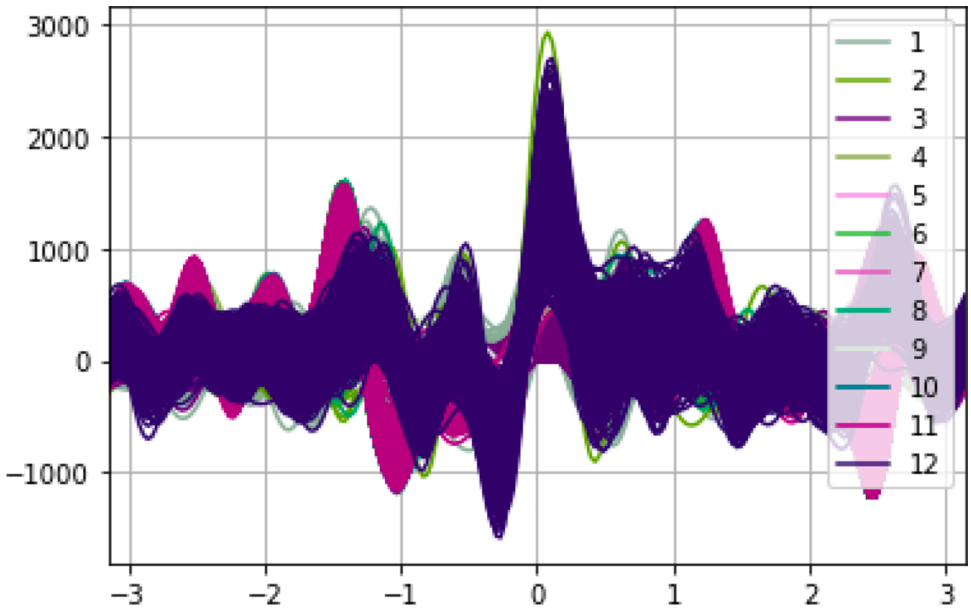


Fig. 4. Andrew curve depicting very high feature space non-linearity for IoT DoS DDoS Attack dataset

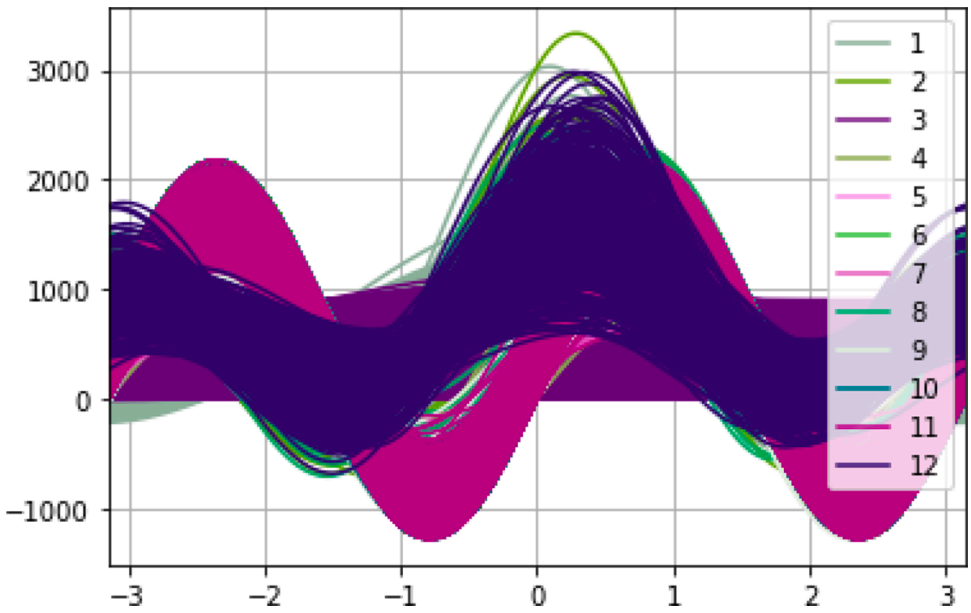


Fig. 5. Andrew curve plot depicting non-linearity of the feature space for IoT DoS DDoS Attack dataset considering 6 dimensions

3.1.3. SDN DDoS dataset

In another case, Andrews's curves are plotted to study SDN DDoS dataset [23,1] . The visualization is shown in Fig. 6. From visualization depicted by Fig. 6, we can understand that the dataset is highly non-linear in nature. In Fig. 6, the class1.0 denotes the normal traffic and class2.0 denotes the attack traffic.

Fig. 7 shows the Andrew curves plot which represents the high non-linearity of the feature space w.r.t the dimensionality reduced SDN DDoS Dataset.

3.1.4. CIC-IDS2017 Intrusion Detection Dataset Friday afternoon logfile

Andrews's curves are plotted by considering the CIC-IDS2017 dataset as shown in Fig. 8. The CIC-IDS 2017 dataset is publicly available by Canadian Institute for Cybersecurity [2,3]. From Fig. 8, it is clear that network traffic in CICIDS2017 dataset is highly non-linear. In the Andrews curves plot shown in Fig. 8, the label 0 represents the benign traffic and labe1 1 indicates attack traffic.

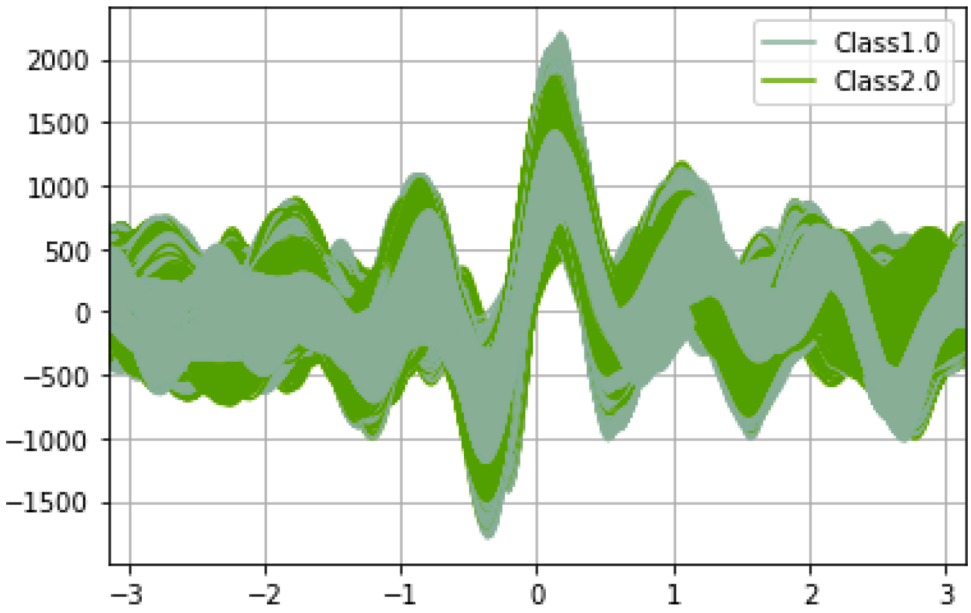


Fig. 6. Andrew Curve showing non-linearity of the feature space for SDN DDoS Dataset

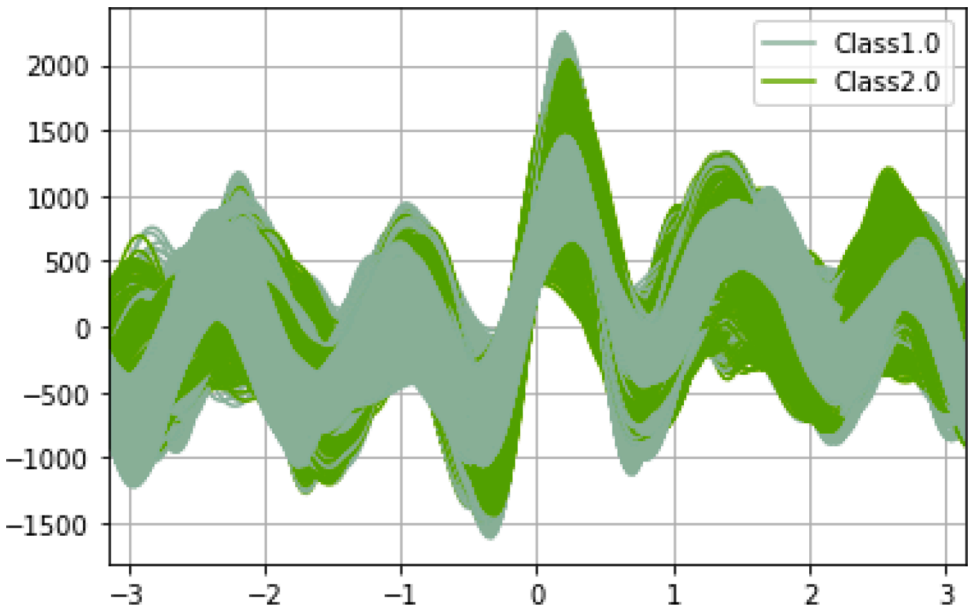


Fig. 7. Plot showing non-linearity of the feature space for dimensionality reduced SDN DDoS Dataset

CICIDS2017 dataset is an IDS dataset which has pcap files of different attacks like DoS, DDoS, Brute Force, XSS, SQL Injection, Infiltration, Port scan and Botnet. This dataset is generated with more than 80 features related to traffic analysis, and finally 78 attributes are finalized out of which one attribute 'Fwd Header Length' is repeated. The dataset contains nine pcap files which are generated from Monday to Friday with testbed architecture. This architecture has two different networks, one is Victim network (network to be attacked) and other is Attack network (to generate attack packets). The attacker network has 1 kali operating system pc, 3 Windows8.1 operating system PC's, a router, and a switch. The victim network architecture has one Domain Controller Server (DC) and DNS server with Windows 2016 server configuration. It also has a webserver with ubuntu 16 and ubuntu 12 operating systems. The attack generation started from Monday and ended by Friday. Captured packets are labelled with corresponding attack names. Monday packet capture file has normal traffic packets, Tuesday morning and afternoon attack traffic is generated by using tools like Hydra, Medusa, Ncrack, Metasploit modules and Nmap NSE scripts and the attack records are distinguished with name Brute Force FTP and Brute Force SSH.

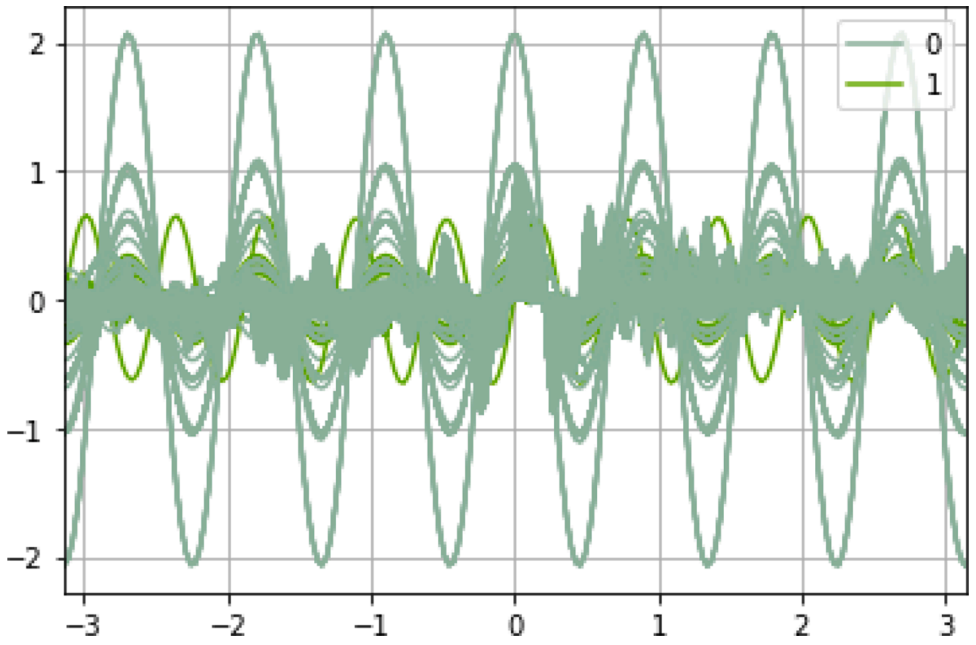


Fig. 8. Andrew curve showing non-linearity of for CICIDS 2017 dataset with 77 attributes

The next section presents experiment results carried out in the present study by considering existing learning models and proposed learning model on IoT DoS and DDoS Attack benchmark dataset mentioned above. The IoT DoS and DDoS Attack dataset is the most recent dataset which is made available at IEEE dataport originally to study the performance of deep learning models. In this paper, we apply the proposed ML model SWASTHIKA to study the performance with respect to attack accuracy and detection rates.

4. Experiment Results and Discussions

In this section, the performance of the proposed method is discussed. At section 4.1, the performance analysis is carried at first by considering the dimensionality reduced traffic data which is obtained using the proposed dimensionality reduction method. Then, multiple linear regression analysis is carried on this traffic data representation. In addition to this, the performance of proposed learning model SWASTHIKA is compared to state-of-art machine learning classifiers by using the reduced dimension network traffic data. All experiments in this research are conducted on system with Intel ® Core ™ i7-9700F CPU @ 3.00GHz 3.00 GHz processor and 16 GB RAM.

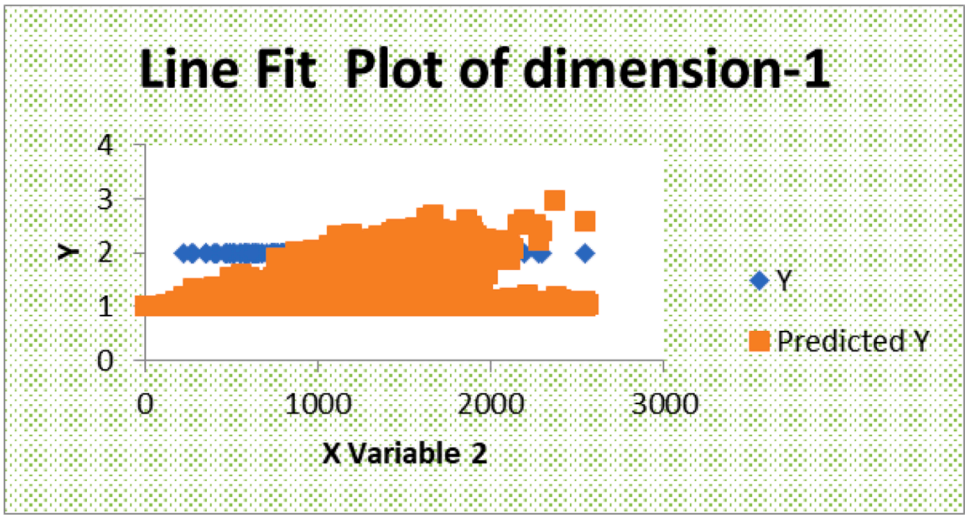
4.1. Regression analysis on IoT DoS and DDoS attack dataset after dimensionality reduction using proposed feature learning approach

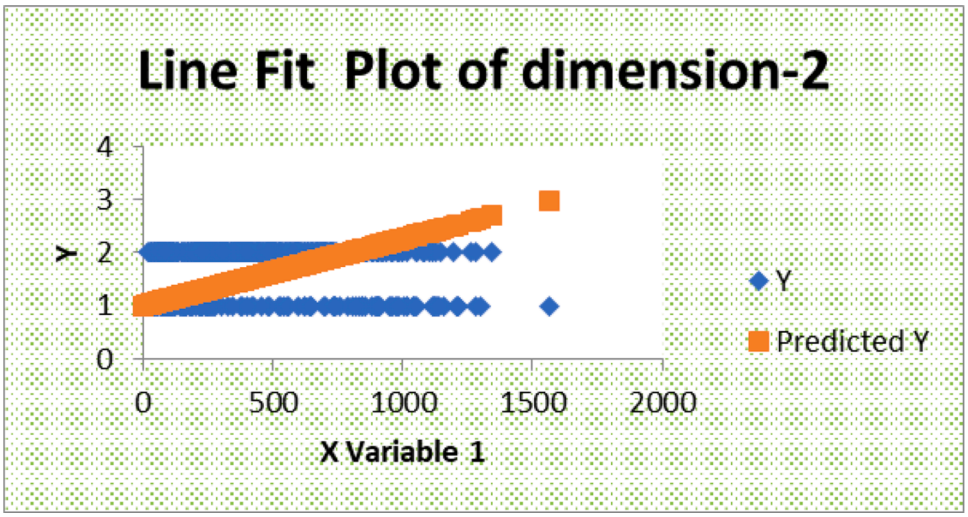
For experiments, we have used the most recent IoT DoS and DDoS Attack dataset available at IEEE Dataport. The original IoT DoS and DDoS attack dataset [13] consisted of 194262 traffic instances in the form of 60 × 60 pixel images.

For experiment analysis, at first these traffic images are reduced to 5 × 5 images. The resulting IoT DoS DDoS dataset hence consisted of 194262 traffic image instances with each image instance of 5 × 5 pixel size. There are 11 attack classes and one normal traffic class in the dataset. For experiment analysis, all these 11 attack classes are combined into one attack class. So, we now have two types of traffic instances, i.e., attack and normal in the modified dataset. To evaluate the performance of proposed dimensionality reduction and machine learning model, each traffic image instance in the dataset is converted into its equivalent multivariate vector of 25 dimensions. So, each of these 194262 traffic instances are now represented as traffic instances of 25 dimensions. All these traffic instances are now fed as input to the proposed dimensionality reduction module in the prediction model. The hyper parameters for dimensionality reduction are threshold (0.95) and deviation (0.5). A deviation value of 0.5 is an unbiased value, hence it is assumed as an initial value. However, in the proposed method, any assumed deviation is finally corrected or updated after the clusters are formed. So, any deviation value between 0 and 1 can be assumed. Preferred values are 0 to 0.5 to avoid bias. In the present case, after the dimensionality reduction is carried, the resulting dimensionality is 2. So, each traffic instance is thus reduced to a 2-dimensionality vector.

Then, the regression model is obtained by considering this 2-dimensionality IoT DoS DDoS dataset and various plots such as residual plot of two dimensions, line fit plot of two dimensions, normal probability plot are plotted. Finally, the overall residual plot and fit chart of the regression model is also obtained.

Fig. 9(a) and Fig. 9(b) shows the residual plot for the first and second dimension of dimensionality reduced IoT DoS DDoS dataset. The line fit plot for two dimensions is shown using Fig. 9(c) and Fig. 9(d). The normal probability plot shown in Fig. 9(e) depicts the





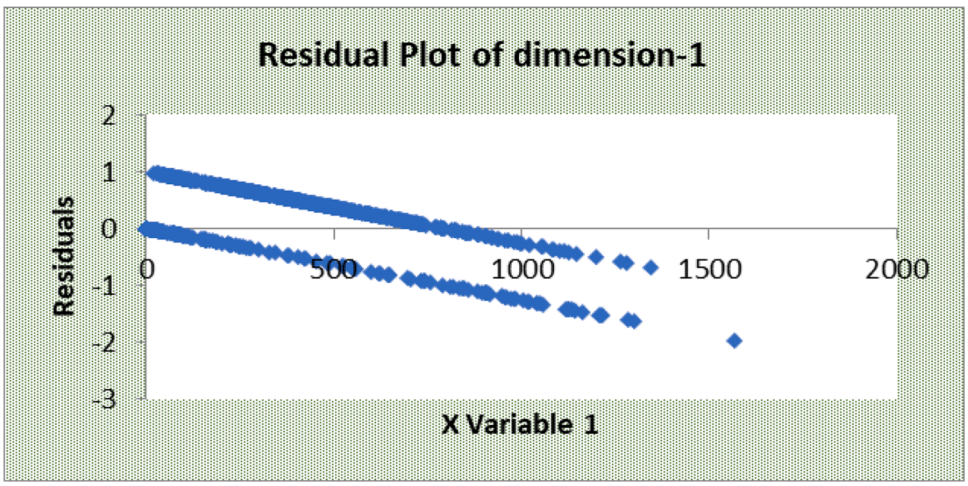


Fig. 9. (a). Residual plot of dimension-1 (b). Residual plot of dimension 2 (c). Line fit plot of dimension-1 (d). Line fit plot of dimension 2 (e). Normal Probability Plot (f). Residual Plot of Regression model (g). Fit chart of Regression model (h). Feature plot after feature transformation using proposed dimensionality reduction

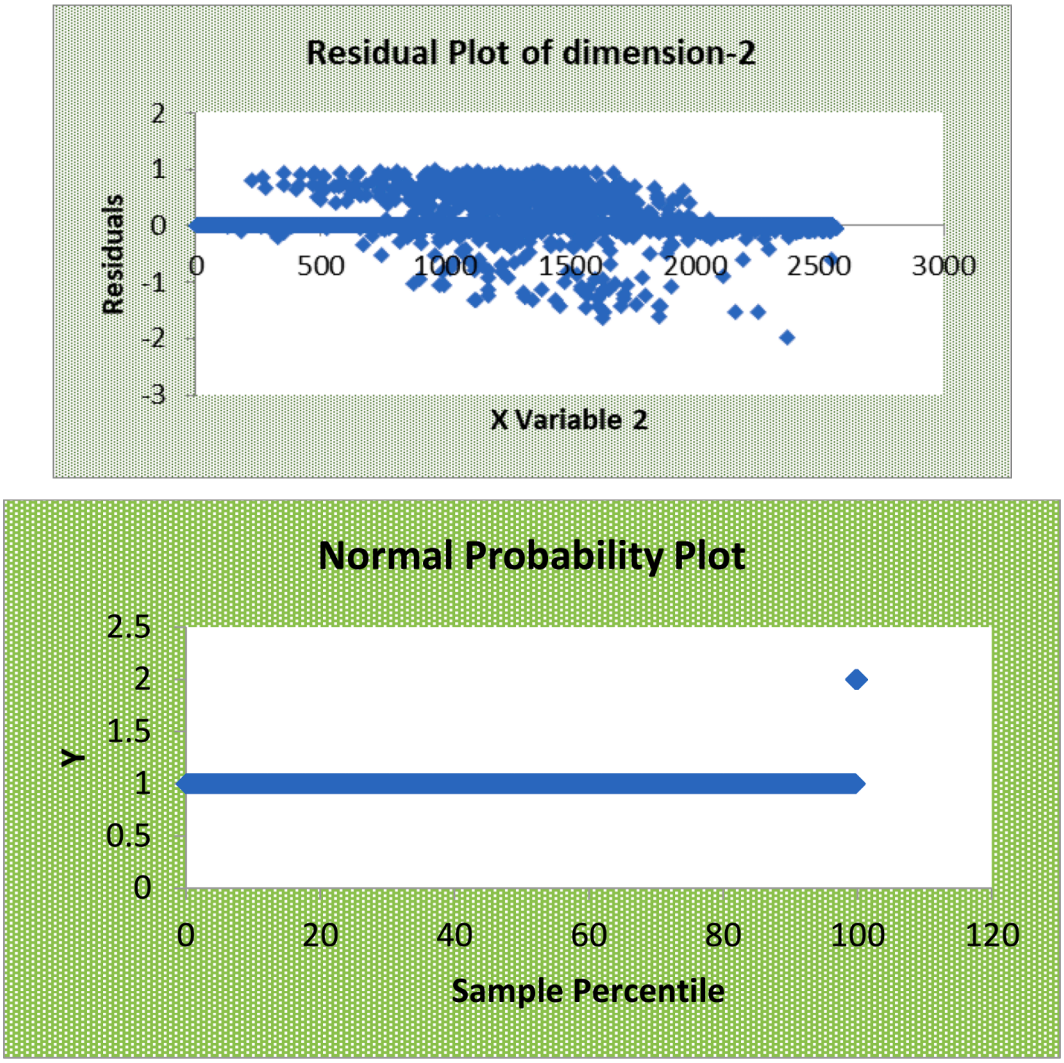


Fig. 9. (*continued*).

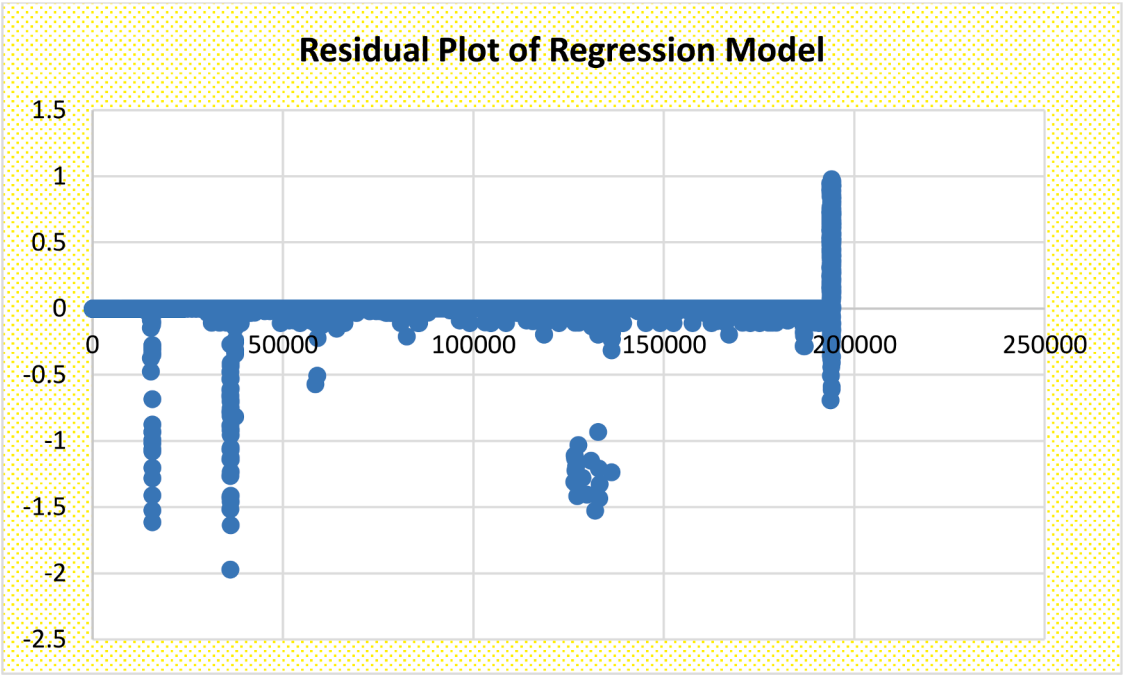
linearity achieved after carrying the dimensionality reduction.

It can be figured out from the residual plot of the regression model shown in Fig. 9(f) that there is no specific pattern that exists in residual plot. This indicates the regression model can be considered. Finally, it is very clear from the fit chart plot of the regression model shown in Fig. 9(g) that the actual and predicted class labels are substantially overlapping. The overlapping of actual and predicted classes indicates that the regression model is performing better w.r.t attack detection. This is visible from the mean absolute percentage error computed for regression model.

In the present case, it is found that the MAPE value of the regression model is obtained as 0.2328 which is very minimal error. The accuracy of the regression model is obtained as 99.76% which is exceptionally better and the same is evident from Fig. 9(f). The overlapping of the actual and predicted traffic instance class in the fit chart indicates that the non-linearity in the dataset is efficiently addressed. Fig. 9(h) shows the feature plot for two features in the reduced dataset. It can be observed from the figure 9(h) that there is minimum overlap which indicates that they are independent and have minimal correlation.

Experiments are also carried by setting threshold to 0.995 and deviation to 0.5 to perform dimensionality reduction. The residual plot, line plot of attributes is depicted using Fig. 10(a) and the overall residual plot for the hyper parameter setting (threshold to 0.995 and deviation to 0.5) is shown in Fig. 10 (b).

The plot in Fig. 10 (b) shows no specific pattern. An observation of the residual plot shows that it is linear. For these settings, ANOVA of Regression model on dimensionality reduced IoT DoS DDoS attack dataset is depicted using Fig. 10(c) and the respective fit chart of the regression model is shown in Fig. 10(d). The regression model accuracy for the IoT DoS DDoS dataset with 2 dimensions is obtained as 99.762% and the MAPE is observed as 0.232%.



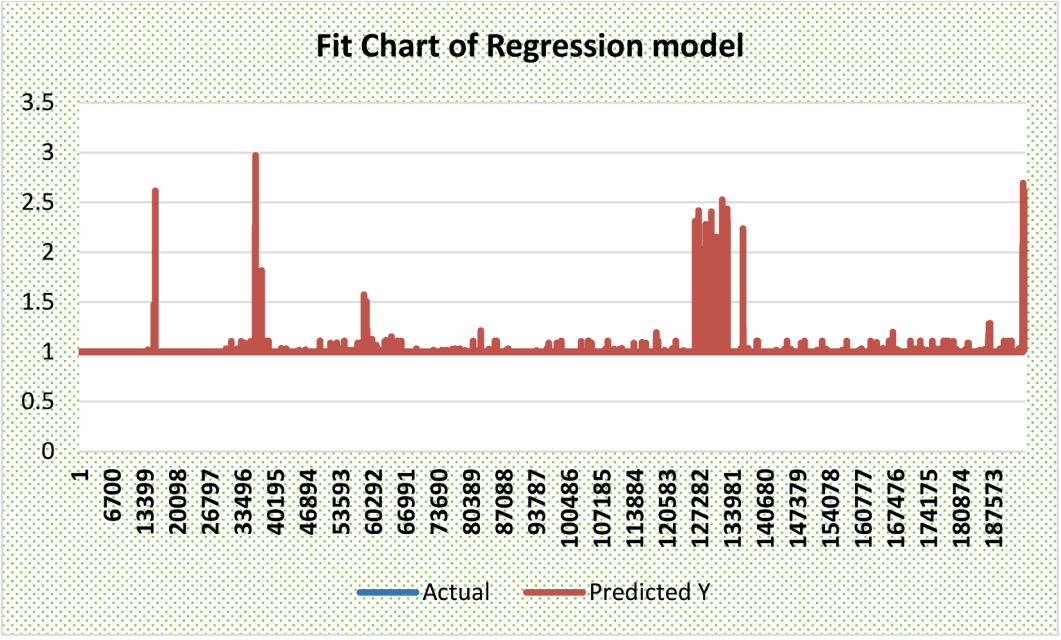


Fig. 9. (*continued*).

4.2. *Performance of Proposed Classifier on IoT DoS DDoS Dataset*

One of the challenging tasks in building an efficient machine learning model is in handling high dimensionality and high non-linearity in multivariate data. It is evident from Fig. 11(a) that feature space of the IoT DoS DDoS attack dataset is highly non-linear. The original dataset has 12 classes (normal and 11 attack traffic instances). For experiment analysis, each traffic image instance is first converted into a traffic image of pixel size 5×5 resulting in 25 feature dimensions. To understand the feature space characteristic, data analysis is carried by plotting Andrew's curve plot. For this, eleven attack classes are first merged into single attack class. Finally, the dataset is modified to have only two types of traffic, i.e., normal and attack. The performance evaluation of the proposed prediction model is done by considering the dataset which is obtained after carrying proposed dimensionality reduction. To achieve this, the proposed network traffic feature similarity function and network traffic similarity function are applied. Fig. 11(b)

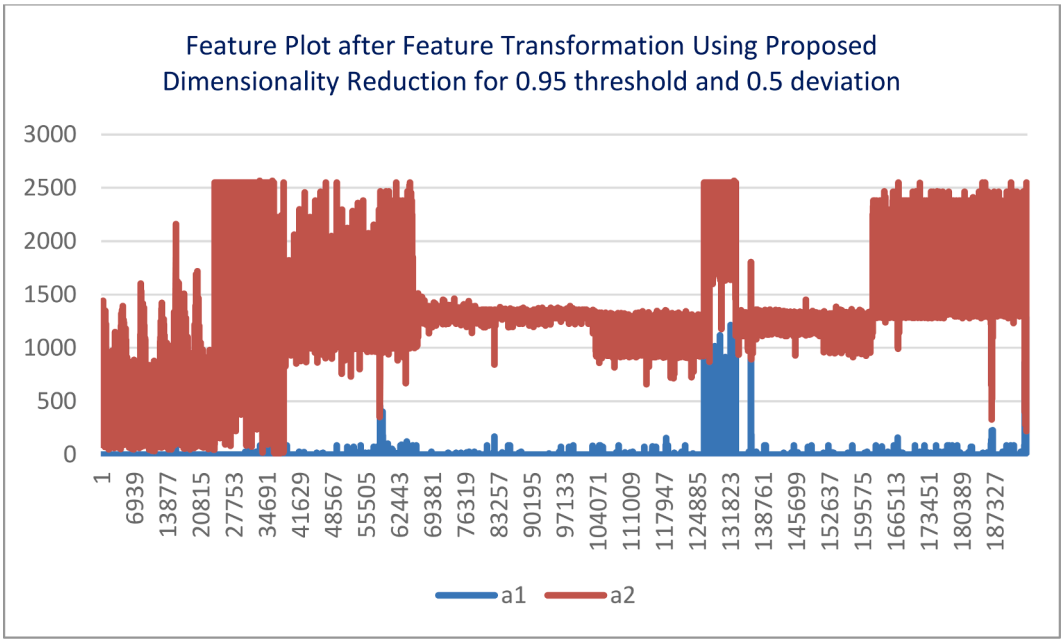


Fig. 9. (*continued*).

depicts the Andrews curve plot for the two class IoT DoS DDoS 2021 attack dataset (CICIDS 2019). It is visible from the plot shown in the figure 11(a), that the feature space of the two-class dataset is highly non-linear.

In addition to this, Andrew's curve plot is also obtained by considering the dataset after carrying dimensionality reduction. After applying the dimensionality reduction using proposed method with hyperparameter setting (similarity threshold - 0.95, gaussian deviation – 0.6), the number of feature dimensions is reduced to one. The Andrews curve plot for the dimensionality reduced traffic is depicted using Fig. 11(b). When compared to Fig. 11(a), the high non-linearity in feature space is slightly reduced in Fig. 11(b) although the feature space is still highly non-linear. On this traffic data, the performance evaluation of proposed prediction model is done.

The IoT DoS DDoS Attack dataset is provided with both training and testing traffic instances. In training set, there are 194262 traffic instances out of which 584 instances are normal traffic and 193678 are attack traffic instances. In testing set, there are 27499 traffic instances out of which 2500 is normal traffic and 25499 are attack traffic instances. When the proposed machine learning classifier model is applied on this dataset, the following test set confusion matrix is obtained as shown in the Table 2.

From the confusion matrix depicted in Table 2, we can compute TP, TN, FP, FN for attack and normal traffic classes. These classifier evaluation parameters are computed from the confusion matrix and are represented using Table 3.

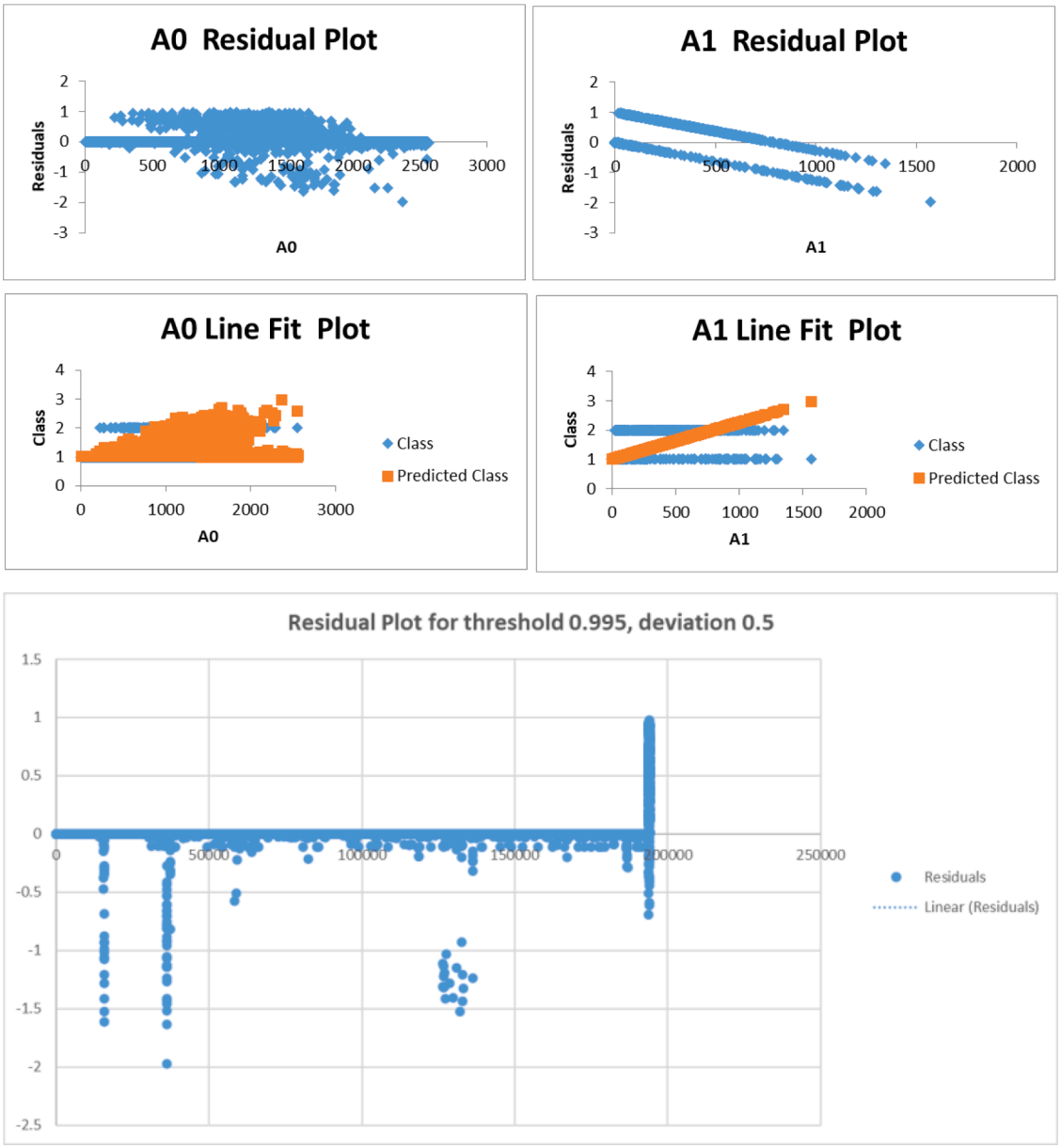
From these values, we can now obtain the remaining evaluation parameters such as Accuracy, Precision, Detection rate, and F-score for attack traffic w.r.t the proposed machine learning model. The percentage accuracy, precision, detection rate and F-score values for attack traffic class are obtained as 90.47%, 90.47%, 99.42% and 0.9499 respectively. The overall percentage accuracy of the proposed learning model for test dataset is obtained as 90.47%.

4.3. Performance of existing ML classifiers on IoT DoS and DDoS attack dataset

For performance analysis of proposed dimensionality reduction method, in this study, we have considered ten state-of-art classifiers. They are Naïve Bayes (C1), Naive Bayes multinomial (C2), RBFC (C3), RBFN (C4), Logistic regression (C5), Simple logistic regression (C6), SMO (C7), BayesNet (C8), J48(C9) and NB-tree (C10). The IoT DoS and DDoS 2021 attack dataset is chosen for performance analysis as this is the most recent benchmark dataset available at IEEE Dataport which satisfies the properties that must be hold by a network traffic to be considered for performance analysis. Performance analysis is carried by considering two test cases (i) IoT DoS and DDoS attack dataset without dimensionality reduction and (ii) IoT DoS and DDoS attack dataset with dimensionality reduction.

4.3.1. IoT DoS and DDoS attack dataset without dimensionality reduction

In this section, experiment results that are obtained by applying various ML classifiers on IoT DoS and DDoS attack dataset without dimensionality reduction are represented in terms of confusion matrices. Confusion matrices obtained for Naïve Bayes (C1), Naive Bayes multinomial (C2), RBFC (C3), RBFN (C4), Logistic regression (C5), Simple logistic regression (C6), SMO (C7), BayesNet (C8), J48 (C9) and NB-tree (C10) classifiers are represented using fig. 12(a), fig. 12(b), fig. 12(c), fig. 12(d), fig. 12(e), fig. 12(f), fig. 12(g), fig. 12 (h), fig. 12(i), and fig. 12(j), respectively.



ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	289.39636	144.698	95984.7	0
Residual	194259	292.84799	0.00151		
Total	194261	582.24435			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
Intercept	1.002346433	0.0002338	4287.7	0	1.001888244	1.002804621	1.001888244	1.002804621
A0	-1.01343E-06	1.637E-07	-6.19122	6E-10	-1.33426E-06	-6.92606E-07	-1.33426E-06	-6.92606E-07
A1	0.001258201	2.872E-06	438.138	0	0.001252573	0.00126383	0.001252573	0.00126383

Fig. 10. (a). Line plot and Residual Plot of attributes considered for building Regression model (b). Residual Plot of Regression model with DR using threshold (0.995) and deviation (0.5) (c). ANOVA of Regression model on dimensionality reduced IoT DoS DDoS attack dataset (d). Fit chart of Regression model on dimensionality reduced IoT DoS DDoS attack dataset

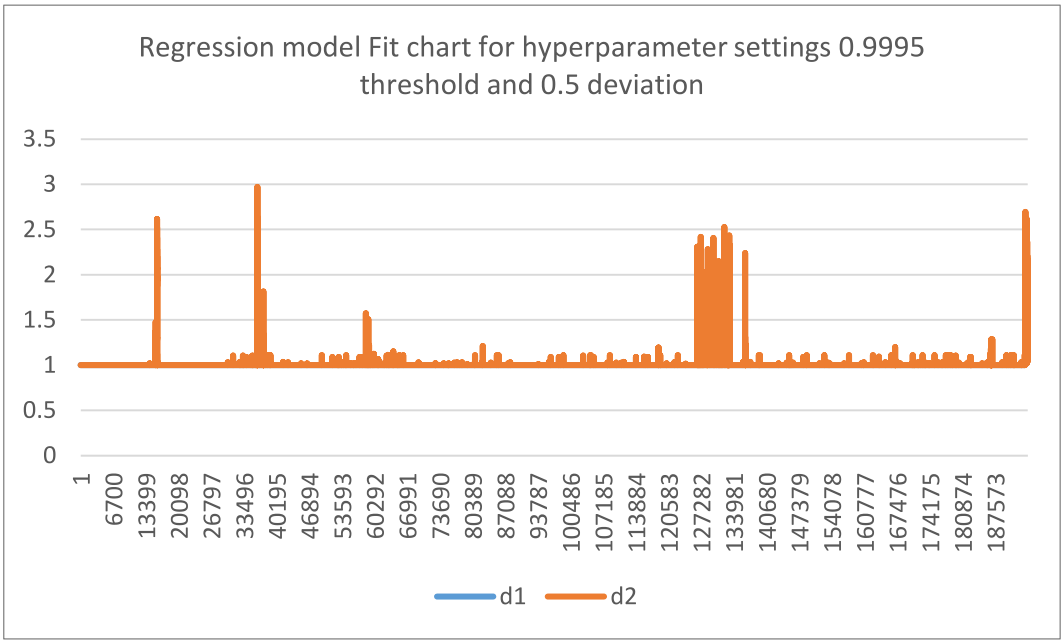


Fig. 10. (*continued*).

4.3.1.1. Naïve Bayes (C1). In the first case, the Naïve Bayes classifier is considered for performance analysis. Experiments are carried by considering training and testing sets. At first, the classifier is trained by considering the training set. The training accuracy, precision, detection rate for attack traffic class is obtained as 98.47%, 99.99% and 98.47% respectively. When test traffic set with 27499 unseen traffic instances is considered, then the test accuracy and detection rates of the naïve Bayes classifier are obtained as 13.86% and 5.25% respectively. In this case, the F-score values for training and test sets are obtained as 0.9923 and 0.0997 respectively. The results show that naïve Bayes classifier has failed to detect unseen attack traffic.

4.3.1.2. Naïve Bayes Multinomial (C2). In the second case, the Naive Bayes multinomial classifier is considered for performance analysis. Experiments are carried by considering training and testing sets. Initially, the classifier is trained using the training set. The training accuracy, precision, detection rate for attack traffic class is obtained as 86.94%, 100% and 86.91% respectively. When test traffic set with 27499 unseen traffic instances is considered, then the test accuracy and detection rates of the Naive Bayes multinomial classifier are obtained as 9.09% and 0% respectively. In this case, the F-score values for training and test sets are obtained as 0.9299 and 0 respectively. The results show that Naive Bayes multinomial classifier has failed to detect unseen attack traffic.

4.3.1.3. RBFC (C3). In the third scenario, RBFC classifier is considered for performance analysis. Experiments are carried by considering training and testing sets. Initially, the classifier is trained using the training set. The training accuracy, precision, detection rate for attack traffic class is obtained as 99.70%, 99.70% and 100% respectively. When test traffic set with 27499 unseen traffic instances is considered, then the test accuracy and detection rates of the RBFC classifier are obtained as 90.91% and 100% respectively. In this case, the F-score values for training and test sets are obtained as 0.9985 and 0.9523 respectively. Though, the RBFC classifier has failed to detect unseen attack traffic, but it has performed better than naïve Bayes and Naive Bayes multinomial classifiers.

4.3.1.4. RBFN (C4). In the fourth scenario, RBFN classifier is considered for performance analysis. Experiments are carried by considering training and testing sets. Initially, the classifier is trained using the training set. The training accuracy, precision, detection rate for attack traffic class is obtained as 99.84%, 99.99% and 99.85% respectively. When test traffic set with 27499 unseen traffic instances is considered, then the test accuracy and detection rates of the RBFC classifier are obtained as 9.43% and 0.38% respectively. In this case, the F-score values for training and test sets are obtained as 0.9992 and 0.0075 respectively. It is observed from experiment results that RBFC classifier has failed to detect unseen attack traffic.

4.3.1.5. Logistic Regression (C5). The performance analysis of the logistic regression model is studied by considering training and testing sets. At first, the classifier is trained by considering the training set. The training accuracy, precision, detection rate for attack traffic class is obtained as 99.85%, 99.90% and 99.94% respectively. When test traffic set with 27499 unseen traffic instances is considered, then the test accuracy and detection rates of the Logistic regression classifier are obtained as 16.25% and 7.96% respectively. In this case, the F-score values for training and test sets are obtained as 0.9992 and 0.1473 respectively. The results show that Logistic regression model has failed to detect unseen attack traffic.

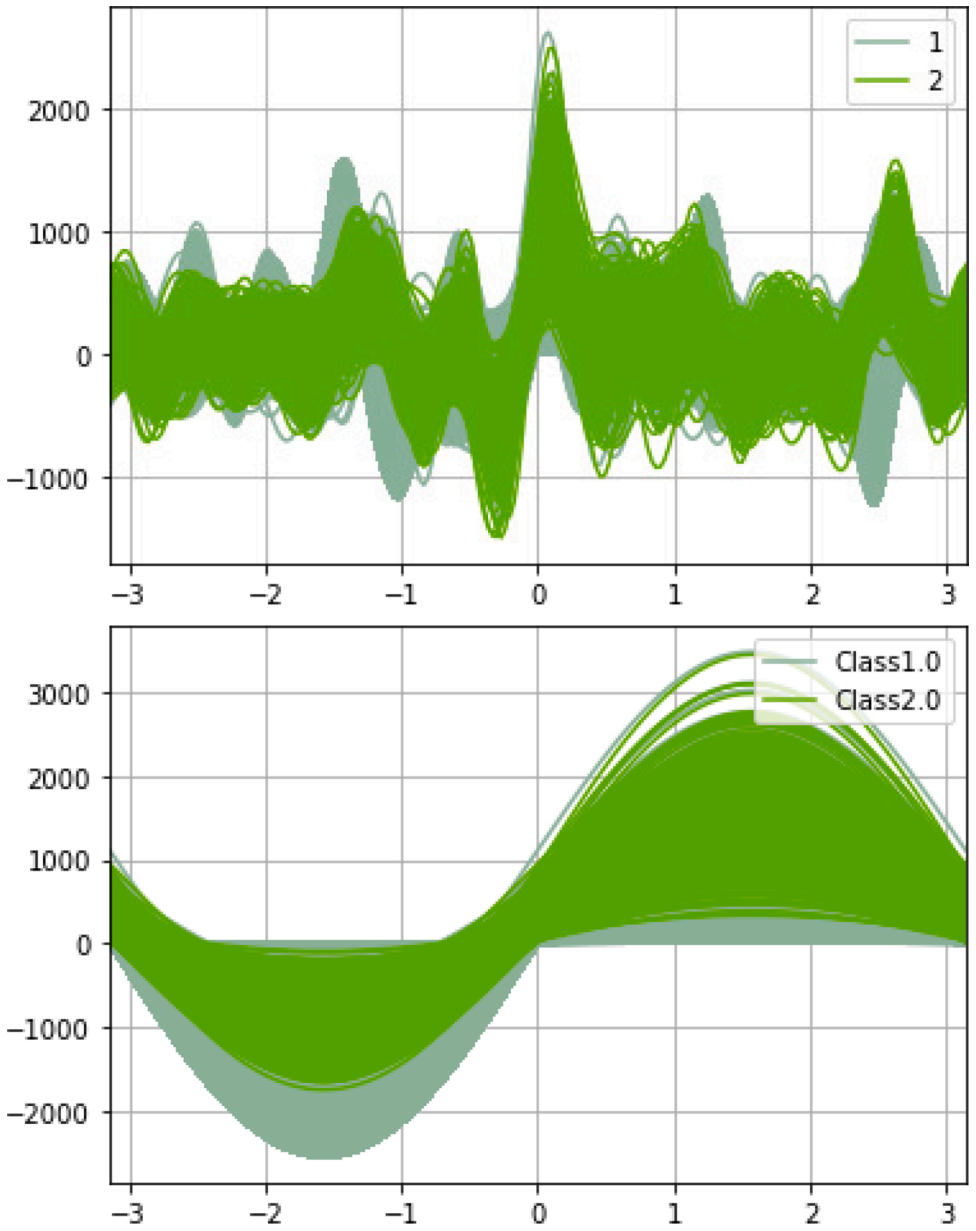


Fig. 11. (a). Andrew Curve Plot showing high non-linearity of feature space for IoT DoS DDoS 2019 dataset before dimensionality reduction (b). Andrew Curve Plot showing non-linearity of feature space for IoT DoS DDoS 2019 dataset after dimensionality reduction using proposed method

Table 2
Confusion matrix

	Attack	Normal
Attack	24855	144
Normal	2476	24

4.3.1.6. Simple Logistic Regression (C6). The performance analysis of the simple logistic regression model is studied by considering training and testing sets. At first, the classifier is trained by considering the training set. The training accuracy, precision, detection rate for attack traffic class is obtained as 99.86%, 99.89% and 99.96% respectively. When test traffic set with 27499 unseen traffic instances is considered, then the test accuracy and detection rates of the simple logistic regression classifier model are obtained as 44.63% and 47.94% respectively. In this case, the F-score values for training and test sets are obtained as 0.9993 and 0.06115 respectively. The

Table 3
TP, TN, FP, FN values for attack traffic using proposed ML model

	Attack Traffic
TP	24855
TN	24
FP	2476
FN	144

	Attack	Normal
Attack	1312	23687
Normal	0	2500

Fig 12(a). Naïve Bayes (C1)

	Attack	Normal
Attack	0	24999
Normal	0	2500

Fig 12(b). Naïve Bayes Multinomial (C2)

	Attack	Normal
Attack	24999	0
Normal	2500	0

Fig 12(c). RBFC (C3)

	Attack	Normal
Attack	95	24909
Normal	0	2500

Fig 12(d). RBFN (C4)

	Attack	Normal
Attack	1990	23009
Normal	21	2479

Fig 12(e). Logistic regression (C5)

	Attack	Normal
Attack	11985	13014
Normal	2211	289

Fig 12(f). Simple Logistic regression(C6)

	Attack	Normal
Attack	9147	15852
Normal	3	2497

Fig 12(g). SMO (C7)

	Attack	Normal
Attack	0	24999
Normal	0	2500

Fig 12(h). Bayesnet (C8)

	Attack	Normal
Attack	47	24952
Normal	0	2500

Fig 12(i). J48 (C9)

	Attack	Normal
Attack	6639	18360
Normal	181	2319

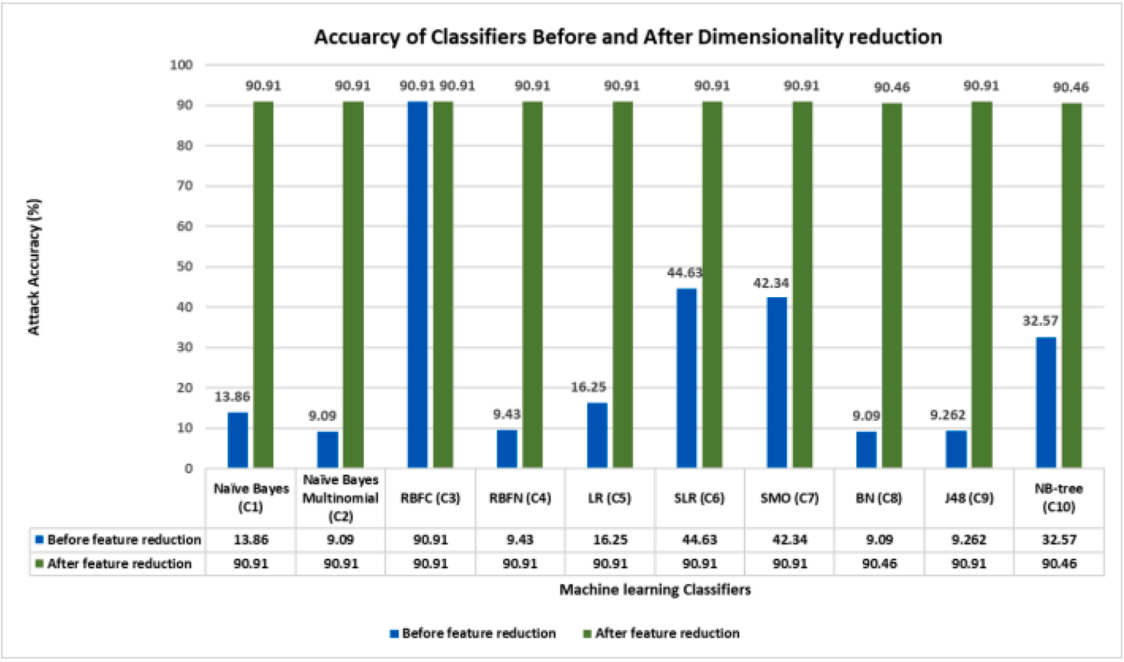
Fig. 12. (a). Naïve Bayes (C1) (b). Naïve Bayes Multinomial (C2) (c). RBFC (C3) (d). RBFN (C4) (e). Logistic regression (C5 (f). Simple Logistic regression(C6) (g). SMO (C7) (h). Bayesnet (C8 (i). J48 (C9) (j). Naïve bayes tree (C10)

results show that the model has failed to detect unseen attack traffic.

4.3.1.7. SMO (C7). Experiment analysis is carried by applying SMO classifier on training and testing traffic. It is observed from the experiment analysis that when training is performed and model is built, then for the attack traffic, the evaluation parameter values are accuracy (99.90%), precision (99.94%), detection rate (99.96%), and F-score (0.9995). When the test traffic is input to the model, then the accuracy, detection rate and F-score values are 42.34%, 36.58% and 0.5357 respectively. The results prove that the SMO classifier failed to detect unseen network traffic efficiently.

4.3.1.8. Bayes Net (C8). Experiment is carried by applying Bayes Net classifier on training and testing traffic. It is observed from the experiment analysis that when training is performed and model is built, then for the attack traffic, the evaluation parameter values are accuracy (99.94%), precision (100%), detection rate (99.94%), and F-score (0.9997). When the test traffic is input to the model, then the accuracy, detection rate and F-score values are 9.09%, 0% and 0 respectively. The results prove that the Bayes Net classifier failed to detect unseen network traffic efficiently.

4.3.1.9. J48 (C9). Experiment is carried by applying J48 classifier on training and testing traffic. It is observed from the experiment analysis that when training is performed and model is built, then for the attack traffic, the evaluation parameter values are accuracy (99.94%), precision (100%), detection rate (99.94%), and F-score (0.9997). When the test traffic is input to the model, then the



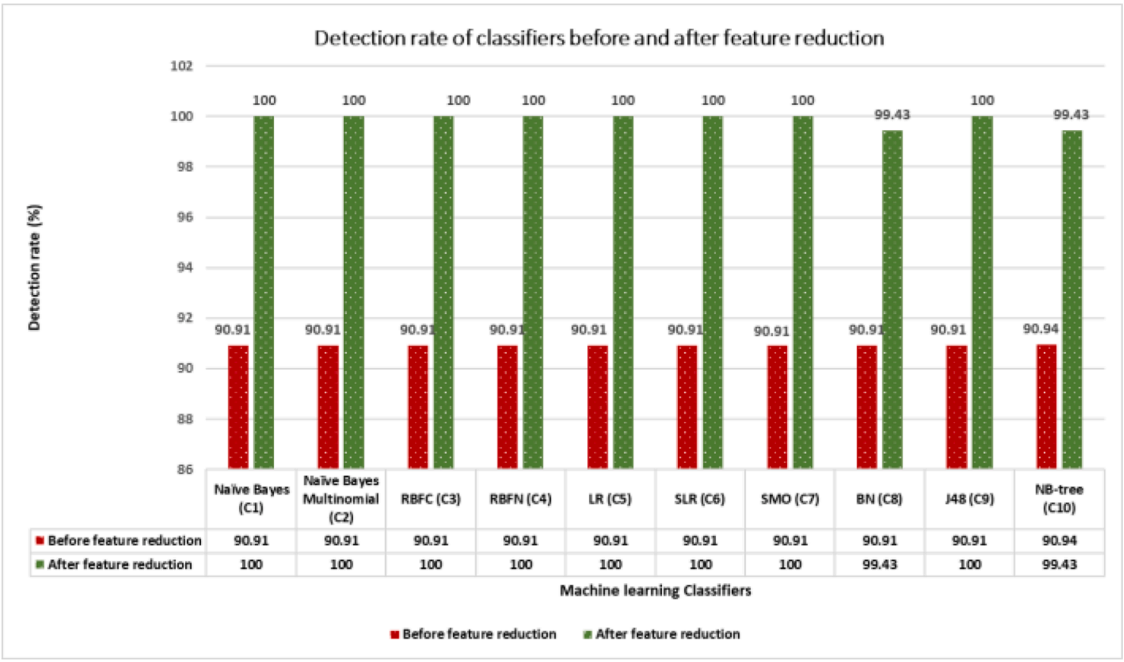


Fig. 13. (a). Test accuracy of state of art classifiers before and after feature reduction (b). Test detection rates of state of art classifiers before and after feature reduction (c). F-score value for various state of art classifiers before and after feature reduction

accuracy, detection rate and F-score values are 9.262%, 0.188% and 0.0037 respectively. The results prove that J48 decision tree classifier failed to detect unseen network traffic efficiently.

4.3.1.10. Naïve bayes tree (C10). Finally, the experiment is also done by applying Naïve bayes tree classifier on training and testing traffic. It is observed from the experiment analysis that when training is performed and model is built, then for the attack traffic, the evaluation parameter values are accuracy (99.99%), precision (99.99%), detection rate (99.82%), and F-score (0.9999). When the test traffic is input to the model, then the accuracy, detection rate and F-score values are 32.57%, 26.55% and 0.4173 respectively. The results prove that Naïve bayes tree classifier failed to detect unseen network traffic efficiently.

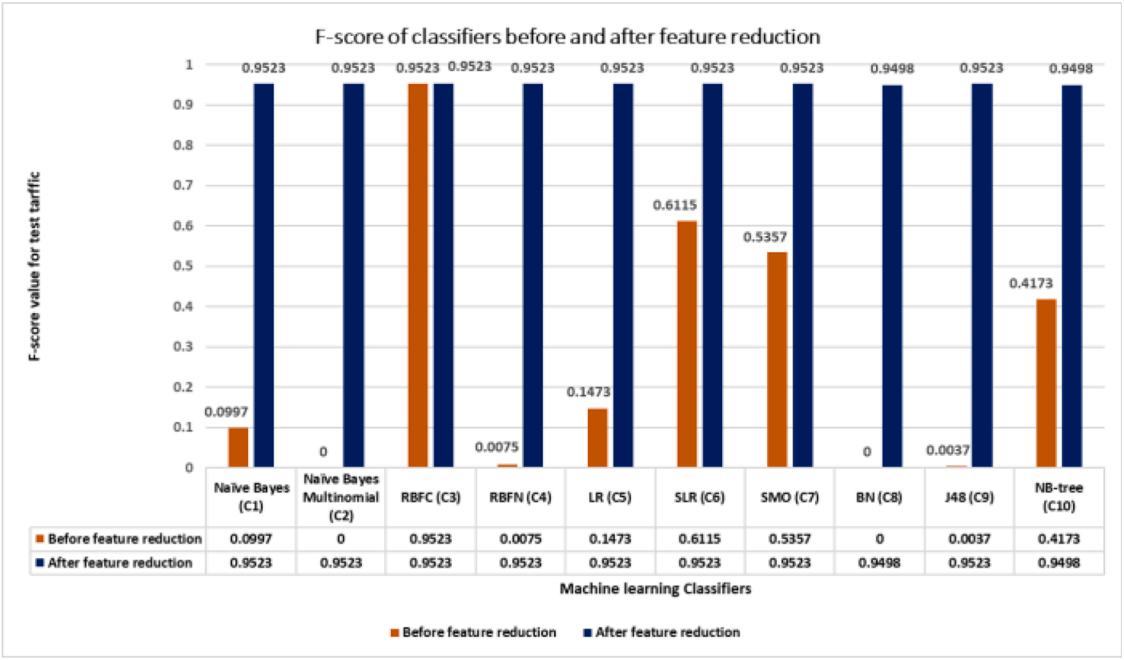


Fig. 13. (*continued*).

4.3.2. *IoT DoS and DDoS attack dataset with dimensionality reduction*

In the second case, experiments are performed by considering network traffic data and carrying feature transformation on traffic data. Thus, this subsection outlines the experiment results which are obtained by carrying feature transformation on IoT DoS and DDoS attack dataset. Initially, the model is trained with 194262 training instances and then the trained model is used to test its performance on 27499 unseen network traffic. The hyperparameter settings for feature reduction includes similarity threshold in Euclidean space (0.95), gaussian threshold (0.427) and gaussian deviation (0.6). Fig. 13(a) shows the accuracy of various classifiers by considering traffic data before and after feature reduction. Fig. 13(b) and Fig. 13(c) shows the detection rate and F-score value obtained for various classifiers. From experiment analysis, it can be verified that the attack accuracy, detection rates and F-score values of Naïve Bayes (C1), Naive Bayes multinomial (C2), RBFN (C4), Logistic regression (C5), Simple logistic regression (C6), SMO (C7), BayesNet (C8), J48(C9) and NB-tree (C10) classifiers are improved when proposed feature reduction is carried on traffic data. For example, consider classifier C1, the accuracy and detection rates of C1 are improved from 13.86% to 90.91% and 5.25% to 100% which is a substantial improvement. For classifier C2, the accuracy and detection rates are improved from 9.09% to 90.91% and 0% to 100% which is a substantial improvement.

Similarly, for classifier C4, the test accuracy and detection rates are increased from 9.43% to 90.91% and 0.38% to 100% respectively. For classifier C5, the test accuracy and detection rates are increased from 16.25% to 90.91% and 7.96% to 100% respectively. For classifier C6, the test accuracy and detection rates are increased from 44.63% to 90.91% and 47.94% to 100% respectively. In case of classifier C7, the test accuracy is increased from 42.34% to 90.91% and detection rate has improved from 36.58% to 100%. The improvement in test accuracy, detection rate and F-score of C8, C9 and C10 classifiers is observed when feature transformation is carried on traffic data. However, for classifier C3, its detection rate and accuracy remained same before and after feature reduction. In case of C8, its accuracy and detection rates are 90.91% and 100% respectively. Thus, the experiment results prove that the proposed feature transformation has improved the performance of the state of art classifiers substantially.

4.4. *Performance Comparison of Proposed Model SWASTHIKA with Benchmark ML Classifiers*

Experiments are conducted by considering the proposed ML classifier model SWASTHIKA on the dimensionality reduced IoT DoS DDoS dataset which is obtained after feature transformation. The hyperparameter settings for feature reduction includes similarity threshold in Euclidean space (0.95), gaussian threshold (0.427) and gaussian deviation (0.6). Fig. 14(a), Fig. 14(b), and Fig. 14(c) depicts accuracy, detection rate and F-Score values obtained for SWASTHIKA and various machine learning classifiers w.r.t attack traffic class. In case of state of art classifiers, the IoT DoS DDoS attack dataset is used without carrying feature reduction. Thus, the performance of state of art classifiers C1 to C10 is evaluated for the traffic data without feature reduction. SWASTHIKA on the other hand carries feature transformation and performs classification by considering dimensionality reduced traffic.

From experiments conducted, it is observed that the accuracy of SWASTHIKA is better when compared to the state of art classifiers namely C1, C2, C4, C5, C6, C7, C8, C9 and C10 which are considered for performance analysis. However, for classifier RBFC (C3), the attack accuracy without feature transformation is achieved as 90.91 for 25 dimensions which is marginally higher than SWASTHIKA

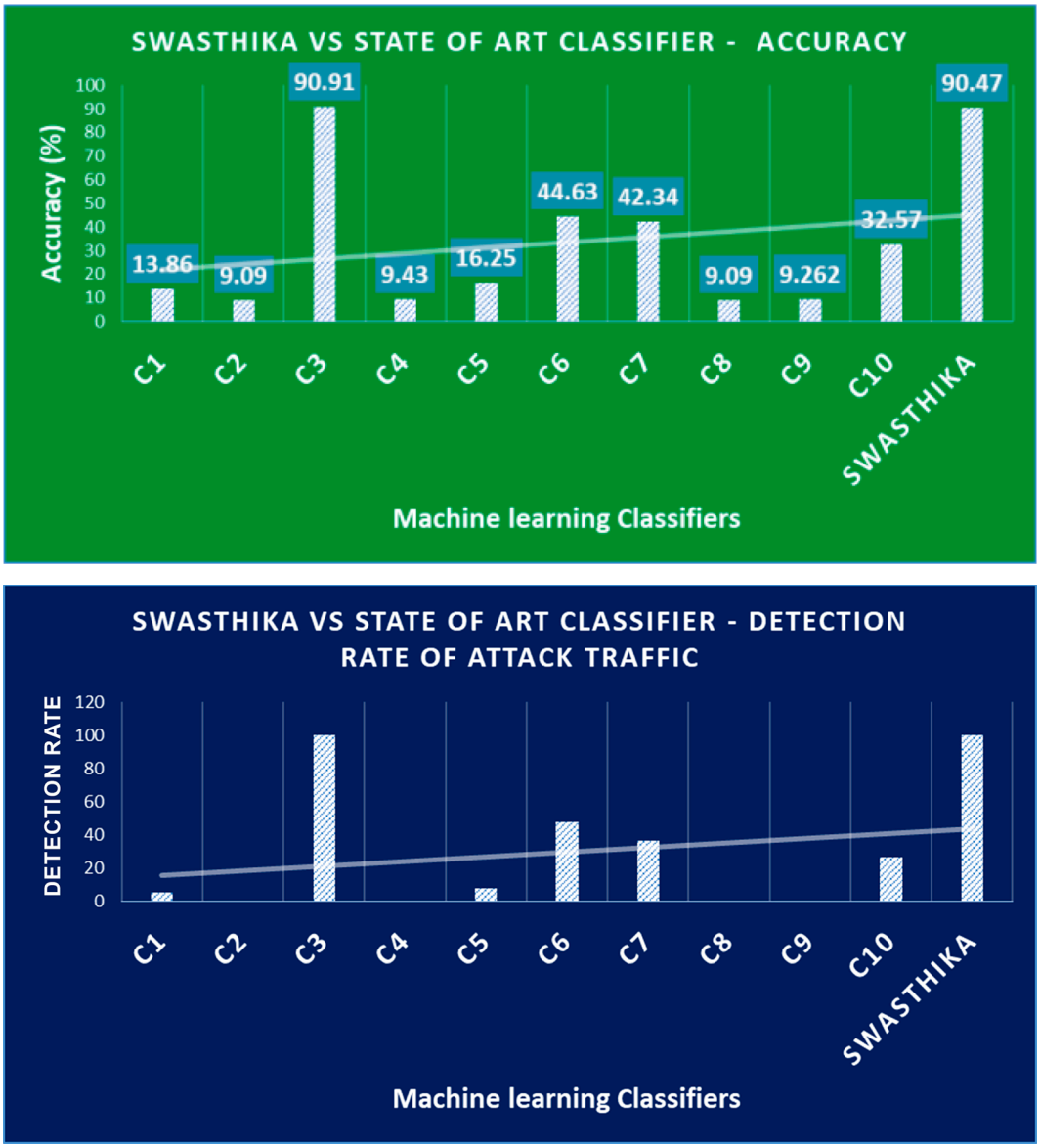


Fig. 14. (a). SWASTHIKA vs other ML Classifiers – Accuracy of attack traffic (b). SWASTHIKA vs other ML Classifiers – Detection rate of test attack traffic (c). SWASTHIKA vs other ML Classifiers – F-score of test attack traffic

by 0.44.

Overall, it is proved from experimental study and analysis that the performance of proposed model SWASTHIKA is substantially better, and it can detect network attacks with better accuracy and detection rates.

5. Conclusions

Recent advances in mezzanine technologies have inspired application of artificial intelligence and machine learning techniques for DDoS attack detection by academia research community and industry. Thus, the fusion of AI and ML based systems and their integration with Industry 4.0 is crucial for the future of industry. An important challenge in Cloud computing and IoT environments is the immediate need to address the security and data availability issues faced by these modern network environments. It is expected that DDoS attacks in cloud can still become challenging in very near future with low-rate DDoS attacks and the high non-linearity of multivariate traffic. The complexity involved in addressing DDoS attack detection problem has thus gained an immediate attention from industry as well as the Cloud, IoT, and SDN researcher community. The integration of AI and ML technologies to build IDS can pave a way to detect complex DDoS and other network attacks. In this research, we have proposed a feature transformation-based

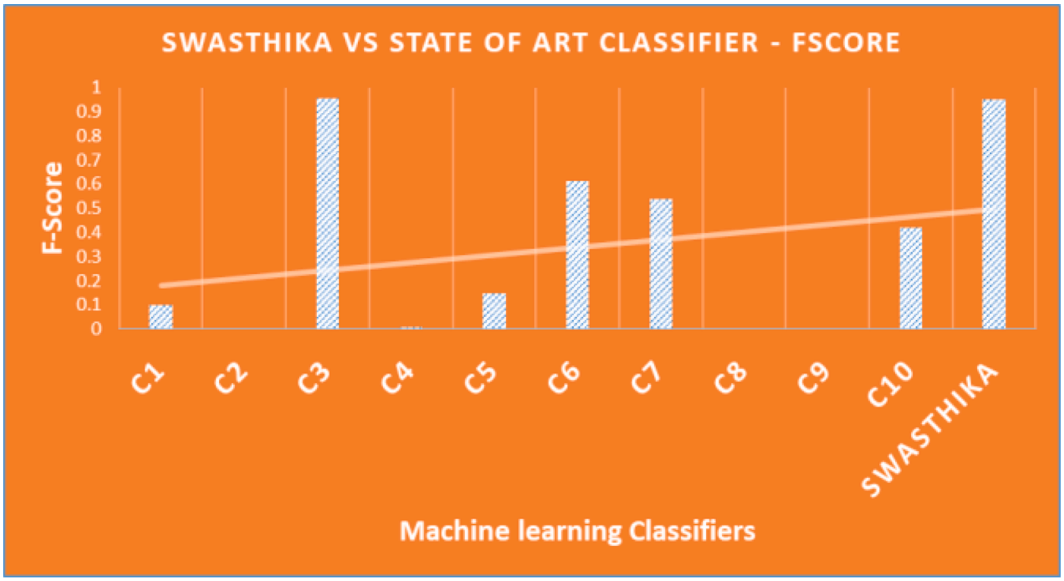


Fig. 14. (*continued*).

machine learning model SWASTHIKA for efficient detection of network attacks. The current study has also presented important visualizations such as Andrews's curves plot for visual analysis of high dimensional multi-variate data, residual plot and fit chart which helps to understand the ML model and its suitability for network attack detection. For performance evaluation of the proposed ML model SWASTHIKA, the recent benchmark dataset namely, IoT DoS and DDoS attack dataset is considered as it satisfies both assessment and evaluation criteria for Intrusion detection. The experimentation results and visual analysis proves that the proposed model for network attack detection has low error rate, higher accuracy and detection rates when compared to other state-of-art classifiers. In future, this work can be extended to come up with ensemble models for detection and classification of various categories of modern network attacks.

Declaration of Competing Interest

None.

Acknowledgements

We thank all the anonymous reviewers of this manuscript who have provided constructive comments for improvements in the present form. Swathi is heartfully thankful to the management of VNRVJIET, Dr. C.D. Naidu, Principal, Dr. N. Mangathayaru, Professor, Dr. G. Suresh Reddy, Professor and Dr. D. Srinivasa Rao, Head of the Department for providing necessary technical facilities to carry the present research. A very special thanks to Aravind Cheruvu, Alumni of VNRVJIET and presently pursuing his M.S at Virginia state university for his continuous support in carrying the present work.

References

[1] Mahjabin T, Xiao Y, Sun G, Jiang W. A survey of distributed denial-of-service attack, prevention, and mitigation techniques. International Journal of Distributed Sensor Networks December 2017. https://doi.org/10.1177/1550147717741463.
[2] Gharib A, Sharafaldin I, Lashkari AH, Ghorbani AA. An evaluation framework for intrusion detection dataset. In: International Conference on Information Science and Security (ICISS), Pattaya, 2016; 2016. p. 1–6.
[3] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," 4th International Conference on Information Systems Security and Privacy (ICISSP), Portugal, 2018.
[4] Sambangi S, Gondi L. A Machine Learning Approach for DDoS (Distributed Denial of Service) Attack Detection Using Multiple Linear Regression. Proceedings 2020;63(1):51. https://doi.org/10.3390/proceedings2020063051.
[5] Sambangi S., Gondi L. (2021) Multi Linear Regression Model to Detect Distributed Denial of Service Attacks in Cloud Environments. In: Singh J., Kumar S., Choudhury U. (eds) Innovations in Cyber Physical Systems. Lecture Notes in Electrical Engineering, vol 788. Springer, Singapore.
[6] Swathi Sambangi and Lakshmeeswari Gondi. 2021. Multiple Linear Regression Prediction Model for DDOS Attack Detection in Cloud ELB. In The 7th International Conference on Engineering & MIS 2021 (ICEMIS'21), October 11–13, 2021, Almaty, Kazakhstan. ACM, New York, NY, USA 9 Pages. 10.1145/3492547.3492567.
[7] Gupta BB, Badve OP. Taxonomy of dos and ddos attacks and desirable defense mechanism in a cloud computing environment. Neural Computing and Applications 2017;28:3655–82.
[8] Sharafaldin I, Lashkari AH, Hakak S, Ghorbani AA. Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy. In: 2019 International Carnahan Conference on Security Technology (ICCST); 2019. p. 1–8. https://doi.org/10.1109/CCST.2019.8888419.

[9] Sharafaldin Iman, Habibi Lashkari Arash, Ghorbani Ali A. An evaluation framework for network security visualizations. Computers & Security 2019;84:70–92. https://doi.org/10.1016/j.cose.2019.03.005.
[10] Aljawarneh SA, Vangipuram R. Garuda: gaussian dissimilarity measure for feature representation and anomaly detection in internet of things. Journal of Supercomputing 2020;76:4376–413.
[11] Kurniabudi D, Darmawijoyo S, Bin Idris MY, Bamhdi AM, Budiarto R. CICIDS-2017 dataset feature analysis with information gain for anomaly detection. IEEE Access 2020;8:132911–21.
[12] Zhijun W, Wenjing L, Liang L, Meng Y. Low-Rate DoS Attacks, Detection, Defense, and Challenges: A Survey. IEEE Access 2020;8:43920–43. https://doi.org/10.1109/ACCESS.2020.2976609.
[13] Hussain F, Abbas SG, Husnain M, Fayyaz UU, Shahzad F, Shah GA. IoT DoS and DDoS Attack Detection using ResNet. In: 2020 IEEE 23rd International Multitopic Conference (INMIC); 2020. p. 1–6. https://doi.org/10.1109/INMIC50486.2020.9318216.
[14] Hussain Faisal, Abbas Syed Ghazanfar, Husnain Muhammad, Fayyaz Ubaid U, Shahzad Farrukh, Shah Ghalib A. IoT DoS and DDoS Attack Dataset. IEEE Dataport August 16, 2021. https://doi.org/10.21227/0s0p-s959.
[15] Yan Q, Yu FR, Gong Q, Li J. Software-Defined Networking (SDN) and Distributed Denial of Service (DDoS) Attacks in Cloud Computing Environments: A Survey, Some Research Issues, and Challenges. in IEEE Communications Surveys and Tutorials 2016;18(1):602–22. https://doi.org/10.1109/COMST.2015.2487361.
[16] Nassif AB, Talib MA, Nasir Q, Albadani H, Dakalbab FM. Machine Learning for Cloud Security: A Systematic Review. in IEEE Access 2021;9:20717–35. https://doi.org/10.1109/ACCESS.2021.3054129.
[17] Zhijun W, Wenjing L, Liang L, Meng Y. Low-Rate DoS Attacks, Detection, Defense, and Challenges: A Survey. IEEE Access 2020;8:43920–43. https://doi.org/10.1109/ACCESS.2020.2976609.
[18] Agrawal N, Tapaswi S. Defense Mechanisms Against DDoS Attacks in a Cloud Computing Environment: State-of-the-Art and Research Challenges. in IEEE Communications Surveys and Tutorials 2019;21(4):3769–95. https://doi.org/10.1109/COMST.2019.2934468.
[19] Yu S, Tian Y, Guo S, Wu DO. Can We Beat DDoS Attacks in Clouds? in IEEE Transactions on Parallel and Distributed Systems Sept. 2014;25(9):2245–54. https://doi.org/10.1109/TPDS.2013.181.
[20] Radhakrishna V, Aljawarneh SA, Kumar PVeereswara, Janaki V. ASTRA - A Novel interest measure for unearthing latent temporal associations and trends through extending basic gaussian membership function. Multimedia Tools and Applications 2019;78(4):4217–65. https://doi.org/10.1007/s11042-017-5280-y.
[21] Radhakrishna V, Aljawarneh SA, Kumar PV, Choo K-KR. A novel fuzzy gaussian-based dissimilarity measure for discovering similarity temporal association patterns. Soft Computing 2018;22(6):1903–19. https://doi.org/10.1007/s00500-016-2445-y.
[22] Vangipuram R, Gunupudi RK, Puligadda VK, Vinjamuri J. A machine learning approach for imputation and anomaly detection in IoT environment. Expert Systems 2020;37:e12556. https://doi.org/10.1111/exsy.12556.
[23] Ahuja Nisha, Singal Gaurav, Mukhopadhyay Debajyoti. DDOS attack SDN Dataset. Mendeley Data 2020:V1. https://doi.org/10.17632/jxpfjc64kr.1.
[24] Lu N, Li D, Shi W, Vijayakumar P, Piccialli F, Chang V. An efficient combined deep neural network based malware detection framework in 5G environment. Computer Networks 2021;189:107932.
[25] Abdel-Basset M, Chang V, Hawash H, Chakrabortty RK, Ryan M. Deep-IFS: intrusion detection approach for industrial internet of things traffic in fog environment. IEEE Transactions on Industrial Informatics 2020;17(11):7704–15.

Further reading

[1] Swathi Sambangi, Lakshmeeswari Gondi, Shadi Aljawarneh, Sreenivasa Rao Annaluri, December1, 2021, "SDN DDOS ATTACK IMAGE DATASET", IEEE Dataport, doi:https://dx.doi.org/10.21227/k06q-3t33.
[2] Laxmi Pranitha Rachamalla, Anusha Akkidasari, Sandhya Madiga, Harshitha Mittapalli, RadhakrishnaVangipuram, November 30, 2021, "CLOUD ATTACK DATASET", IEEE Dataport, doi:https://dx.doi.org/10.21227/05ep-zk84.
[3] https://www.unb.ca/cic/datasets/ids-2017.html.

Swathi Sambangi is a Research Scholar at GITAM Institute of Technology, GITAM (Deemed to be University), Visakhapatnam and presently serving as an Assistant Professor in the Department of Information Technology at VNR Vignana Jyothi Institute of Engineering and Technology, Telangana, India. She is awarded B. Tech in Information Technology and Master of Technology in Software Engineering from JNTU Kakinada. She is a member of ACM and has ten years of academic teaching experience. Swathi has presented and published research papers at several international conferences and journals. Her areas of research interest are Algorithm design, Cloud Security and Machine learning.

Lakshmeeswari Gondi is an Associate Professor at GITAM Institute of Technology, GITAM (Deemed to be University), Visakhapatnam, India. She is awarded M. Tech in 2009 and Ph. D in 2013 from GITAM University. Several research scholars are working towards their doctoral degree under her esteemed guidance. She has twenty years of academic teaching and fifteen years research experience. She has to her credit several publications in international journals and conferences. Her areas of interest include Data mining, Cloud Computing, Network Security, Visual Cryptography and IOT.

Prof. Shadi Aljawarneh is an ACM Senior member, IEEE member and a full professor of Software Engineering, at Jordan University of Science and Technology. Also, he is a visiting professor at Concordia University, Canada. He holds a BSc degree in Computer Science from Jordan Yarmouk University, a MSc degree in Information Technology from Western Sydney University and a PhD in Software Engineering from Northumbria University-England. Aljawarneh has presented at and been on the organizing committees for a high number of international conferences and is a board member of the International Community for ACM, IEEE, Jordan ACM Chapter, ACS, and others. A good number of his papers have been selected as "Best Papers" at conferences and journals. Also, he has served as a conference chair, TPC chair for a good number of international conferences. Furthermore, Aljawarneh is an associate editor at Computer & Electrical Engineering Journal, Elsevier, associate edior at IEEE ACCESS, academic editor at PeerJ Computer Science and guest editor for many journals special issues.