Moment Distances from Robust Subspace for Botnet Detection

Thiago P. B. Vieira[a], Eduardo S. C. Vilaça[a], João Paulo C. L. da Costa[a,b,c], Éder S. Gualberto[a], Rafael T. de Sousa Júnior[a]

[a] *Department of Electrical Engineering, University of Brasilia (UnB), 70910-900, Brasília-DF, Brazil*
[b] *Institute for Information Technology, Ilmenau University of Technology, Ilmenau, Germany*
[c] *Fraunhofer Institute for Integrated Circuits IIS, Erlangen, Germany*

Abstract

The U.S. government estimates that malicious cyber activity cost the U.S. economy between US$57 billion and US$109 billion in 2016. Accenture argues that 9% of survey respondents believe breakthrough technologies, like artificial intelligence, machine learning and user behavior analytics, are essential for securing organizations. e imbalanced data can compromise the performance of standard learning algorithms, creating bias or unfair weight to learn from the majority class and reducing detection capacity of anomalies. Some widely adopted algorithms for anomaly detection assume a Gaussian distributed data for legitimate observations, but this assumption may not be observed in network traffic, which is usually characterized by skewed and heavy-tailed distributions. We propose the Moment-based Robust Principal Component Analysis (m-RPCA) for botnet attack detection, which is a framework based on distances between contaminated observations and moments computed from a robust subspace learned by RPCA, to detect anomalies and network attacks. We evaluate m-RPCA on simulated data and on CTU-13 data set, and results show that m-RPCA can improve the botnet attack detection and the anomaly detection on skewed data, in comparison to anomaly detection algorithms.

Keywords: Anomaly Detection, Botnet Attack Detection, Skewed Data, Imbalanced Data, Robust Principal Component Analysis (RPCA).

1. Introduction

According to [1], 37.9% of all Internet traffic of 2018 was not from human activities, but from bots, can be classified as good or bad bots, which accounted for 20.4% of all website traffic. A botnet collection of bots or systems for executing automated tasks, often in the form of compromised hosts, including desktops, mobile devices or things of the Internet of Things (IoT). Some examples of botnets include Cyclone, Mirai, Nitol and Sentry MBA [2].

ccording to Hevesi [3] in a Gartner's publication, botnets scan the Internet looking for unprotected IoT devices, which are projected to be 25 billion devices in 2021. Hevesi argues that if 1% of IoT devices is added to a botnet and had access to 5G, the potential throughput would be 50 Gbps in a flooding attack.

Distributed attacks organized by botnet l ncreased and demanded the development of counter measures to detect and avoid unknown attacks or even to deal with adversarial changes of behavior, location and other patterns [4, 5]. To face the adversarial model, network attacks and counter measures of attackers to avoid detection, it is possible to adopt unsupervised or semi-supervised approaches for network anomaly detection, by means of behavioral analysis, where it is not necessary known anomalies for training models [6].

Anomalies in the context of network traffic can be hard to identify and separate from legitimate data due to the rare occurrences of anomalies in comparison to legitimate events. Additionally, widely adopted algorithms for anomaly detection assume a Gaussian or symmetric distributed data [7], howev his assumption may not be observed in some real world problems [8], such as the case of network traffic analysis, where network traffic features are usually more characterized by skewed and heavy-tailed distributions [9].

Findi of Benson *et al.* [8] indicate that certain positive skewed and heavy-tailed distributions can model data center switch traffic, and highlights a difference between the data center environment and the wide area network, where the long-tailed Pareto distribution typically shows the best fit [8], eon-Garcia [9] also argues that Pareto distribution has been found to capture the behavior

of many quantities of interest in the study of Internet behavior. Moreover, Benson *et al.* [8] observes that the Lognormal distribution is the best fit to model arrival processes in a data center.

The findings presented by Benson *et al.* [8] and Leon-Garcia [9] show that the skewness and heavy-tailed distributions may be important for network traffic analysis, and can motivate researches to evaluate the impact of skewed data into algorithms that rely on Gaussian data for network anomaly detection. Additionally, the fitting of network traffic to skewed and heavy-tailed distributions can indicate opportunities to exploit characteristics of the skewness and heavy-tailed distributions to improve classifiers for network anomaly detection.

Network anomaly detection problems are usually characterized by imbalanced data [10]. However, learning algorithms for imbalanced data has been a challenging topic, considering that imbalanced data can compromise the performance of most standard learning algorithms, creating bias or unfair weight to learn from the majority class and reducing the detection capacity [11]. Therefore, learning methods for imbalanced and skewed data have attracted attention of researchers [12].

Robust subspace learning has been attracting a growing attention of researchers aiming the development of network attack detection systems that rely on behavioral analysis [13, 14]. An outlyingness-approach based on a robust estimator of skewness, combined with robust estimators of location and scale, can be able to flag the outlying measurements [12]. According to Hubert *et al.* [12], when the same methodology would be used with non-robust estimators of location, scale and skewness, the outlyingness-values would be affected by the outliers such that the outlying group could be masked.

Considering that the skewness of anomalous and legitimate traffic can highlight features for improving anomaly detection and network attack detection in imbalanced data, and considering that the distance between robust estimates can be used for network attack detection, we propose the Moment-based Robust Principal Component Analysis (m-RPCA), which is a framework based on distances between moments computed from a robust subspace learned by Ro-

bust Principal Component Analysis (RPCA) from contaminated observations, to detect anomalies from skewed data and network traffic.

The m-RPCA relies on a robust subspace computed from supposed legitimate observations, for estimating the moments to be used for distance analysis. The anomaly detection from contaminated observations evaluate the Mahalanobis distance between the robust moments and new contaminated observations, in a semi-supervised fashion. The m-RPCA can also be computed as an unsupervised algorithm, with subspace learning from the same contaminated data that is the target of the anomaly detection analysis.

We evaluate the accuracy of the m-RPCA for anomaly detection on simulated data set, with skewed and heavy-tailed distributions, and for botnet attack detection on CTU-13 data set [15], which is a large data set of legitimate, background and botnet traffic that has been adopted to deal with the lack of up-to-date real-world data sets for network attack detection [16]. The experimental evaluation compares m-RPCA to standard and widely adopted algorithms for anomaly detection, which are based on clustering and statistical approaches, and to ROBPCA-AO [17], which is an anomaly detection method that relies on robust estimates with adjusted outlyingness based on robust skewness.

The main contributions of this work are a novel semi-supervised and unsupervised method for anomaly detection in skewed and imbalanced data, with results showing improvements in experimental evaluation on simulated skewed and heavy-tailed data and on real data set with legitimate and botnet traffic.

This paper is organized as follows. In Section 2, a literature review about network anomaly detection, botnet detection, and imbalanced learning is conducted. We present in Section 3 the data model and the evaluated data set. In Section 4 it is described the proposed framework for network attack detection. In Section 5 we discuss the experimental validation, the results for simulated scenarios and the results for botnet attack detection on CTU-13 data set. Finally, in Section 6 we draw the conclusions and the suggestions for future work.

2. Literature Review

Network Anomaly detection has emerging as an important approach for securing communication networks and to deal with the increasing number of network attacks. Bhuyan *et al.* [18] provide an overview of facets of network anomaly detection, present attacks normally encountered by network intrusion detection systems, and categorize existing network anomaly detection methods and systems based on the underlying techniques. Ahmed *et al.* [19] present an analysis of four major categories of anomaly detection techniques, which include classification, statistical, information theory and clustering approaches. Moustafa *et al.* [6] discuss aspects of anomaly-based Network Intrusion Detection Systems (NIDSs), describing details of cyber-attacks and new solutions for anomaly detection, and provides a benchmark data set for training and validating approaches for network anomaly detection.

According to [1], 37.9% of all Internet traffic of 2018 was not from human activities, but from bots, that can be classified as good bots or bad bots. A botnet is a network of bots, that are compromised machines under the influence of malware (bot). The botnet is commandeered by a botmaster and used as resource for attacks, such as distributed denial-of-service (DDoS) attacks, and fraudulent activities, such as spam, phishing, identity theft, and information ex-filtration. We refer to [19] and [6] for an overview of network attacks. The botmaster coordinate a botnet through a command and control (C&C) channel where bots receive commands and synchronize attacks and fraudulent activities. Centralized C&C structures using the Internet Relay Chat (IRC) protocol have been utilized by botmasters for a long time, but other protocols, such as HTTP, and architectures, such as Peer-To-Peer, have also been adopted [20].

Acarali *et al.* [21] surveys network-based detection approaches for HTTP-based botnets, and discuss the traffic-based features used to detect bot traffic and presents an abstraction of the main types of features related to protocols and OSI layers. Wang *et al.* [5] present an analysis based on 50,704 different Internet DDoS attacks originated of 674 botnets from 23 different botnet families

with a total of 9,026 victim belonging to 1,074 organizations in 186 countries. Their analysis reveals that geolocation of the attacking sources follows patterns and enables source prediction, and highlights that multiple attacks to the same target also exhibit strong patterns of inter-attack time interval, also presents that there is a trend for different botnets to launch DDoS attacks targeting the same victim, simultaneously or in turn.

The BotHunter was proposed by Gu *et al.* [22] to detect the infection and coordination of botnets by matching sequence model, through a correlation approach for detecting stages of the infection process. Gu *et al.* [20] presented the BotMiner, which aims to detect groups of compromised machines that are part of a botnet. BotMiner monitors communications that may suggest C&C or malicious activities, and finds a coordinated group pattern by means of clusters of similar communication activities, clusters of similar malicious activities, and performs cross cluster correlation to identify the hosts that share both similar communication patterns and similar malicious activity patterns.

Khattak *et al.* [23] proposed BotFlex, which is a network-based tool for botnet detection, composed by a Complex Event Processing (CEP) engine and on a correlation framework that continuously receives events and correlates them according to rules. BotFlex's results are compared to BotHunter [22], but the evaluation relies in an own and not public data set.

Several approaches for network attack detection uses the KDD 99 [19, 16, 18] data set for accuracy and performance evaluation, due to their public availability and labeled traffic. Even though the KDD 99 are criticized by the generation procedure and the risk of over-estimations of anomaly detection due to data redundancy, it still represents one of the few publicly available labeled data set in use by researchers [16, 18]. NSL-KDD [24] data set is the refined version of the KDD 99 that removed the redundant data records, to avoid biased classifications. However, NSL-KDD maintain the limitations of the KDD 99 regarding volume and lacks on reproduction of recent network traffic and attacks.

Garcia *et al.* [15] compare three botnet detection methods by means of a simple and reproducible methodology, by a good data set and by a new error

metric. This paper evaluates data sets for network anomaly detection, and surveys approaches for botnet detection, and proposes two methods (BClus and CAMNEP) for botnet detection, comparing results to BotHunter [22].

Considering the lack of available labeled data sets, Garcia *et al.* [15] proposes the CTU-13 data set, which is composed by attack, legitimate and background labeled data, in an imbalanced distribution and according to real network traffics. The authors recommend scenarios for training and testing to avoid the use of traffic from a botnet family for training and testing, aiming to ensure that the evaluated methods can generalize and detect new behaviors. Taking into account the adoption of unsupervised or semi-supervised approaches for anomaly detection, if adopted the training and testing approach proposed by Garcia *et al.*, some botnet malwares wouldn't be tested, since in the author's proposal, some botnets are present only for training.

Wang and Paschalidis [4] proposed a botnet detection approach based on anomaly and community detection, aiming for detecting botnets and identifying bots before the botnet becomes active. The first stage detects anomalies by leveraging large deviations of an empirical distribution. The second stage detects the bots using ideas from social network community detection in a graph that captures correlations of interactions among nodes over time. This work is compared to the BotHunter [22] for the CTU-13 data set [15].

According to Ringberg *et al.* [25], traditional PCA-based anomaly detection models are not suitable for anomaly interpretation, as they judge whether a data instance is an anomaly or not based on the length of its projection on the abnormal subspace spanned by the less significant principal components, and there is no direct mapping between PCA's dimensionality-reduced subspace and the original subspace [25]. To overcome the above mentioned limitations, some approaches based on PCA have been proposed for network anomaly detection.

Callegari *et al.* [26] proposed a PCA-based method for identifying the network traffic flows responsible for an anomaly detected at the aggregate level, by means of a separation of legitimate and anomaly observations according to principal components (legitimate) and remaining (anomalies). Lee *et al.*

[27] presented OverSampling PCA (osPCA), which allows one to determine the anomaly of the target instance according to the variation of the resulting dominant eigenvector obtained by similarity analysis and over sampling. Vieira *et al.* [28] proposed a framework that applies Model Order Selection (MOS) for detection of time frames under attack and uses similarity analysis to extract details and detect the time and ports under attack.

The problem of PCA or subspace learning for outlier corrupted data is called Robust Principal Component Analysis (RPCA) or robust subspace learning [29, 30]. RPCA aims to be resilient to outliers by means of a robust subspace learning [30] for outlier corrupted data, decomposing a given data matrix $\boldsymbol{X}$ into the sum of a low rank matrix $\boldsymbol{L}$, whose column subspace gives the principal components, and a sparse matrix $\boldsymbol{S}$, which refers to outliers' matrix. We refer to [31] and [30] for more details regarding robust subspace learning.

RPCA has been mainly applied to computer vision, in problems of robust subspace tracking and robust subspace recovery. However, RPCA has also been adopted for general outlier detection [17, 12, 32, 14, 33] and for anomaly detection on network traffic [13]. ROBPCA-AO [17] intends to identify outliers using PCA from robust estimates of mean and covariance matrix, to reduce the data dimensions and plotting the orthogonal distances versus the robust score distances, to flag an outlier map. However, ROBPCA flags many points as outlying when the original data is skewed. Therefore, Hubert *et al.* [12] proposed ROBPCA-AO, which improves ROBPCA for problems with skewed data, by means of an adjusted outlyingness based on robust skewness. Hubert *et al.* [12] evaluated ROBPCA-AO for real and simulated data set, but it is not clear if this method was evaluated for very skewed and imbalanced data set, such as in the network attack detection problem.

Robust subspace learning has received a growing attention of researchers aiming the development of network anomaly detection systems [12, 13, 14], considering outlier-robust methods and sparse-corruption methods [31]. Pascoal *et al.* [13] proposed an approach based on a robust mutual information estimator for feature selection and based on RPCA for outlier detection in internet traf-

fic. The anomaly detection proposed by Pascoal *et al.* [13] is an unsupervised approach that estimate the first k robust principal components, calculate the score and the orthogonal distances, calculate the thresholds and classify new observations accordingly.

Zhou and Paffenroth [14] proposed the Robust Deep Autoencoders (RDA), which the central idea is that a RDA inherits the non-linear representation capabilities of autoencoders combined with the anomaly detection capabilities of RPCA. Considering that outliers and noise may reduce the quality of representations discovered by deep autoencoders, the proposed model isolates noise and outliers in the input by means of a RPCA, and the autoencoder is trained after this isolation. RDA was evaluated by the authors for the MNIST data set.

Benson *et al.* [8] conducted an empirical study of the network traffic in 10 data centers belonging to three different types of organizations, including university, enterprise, and cloud data centers. Findings of Benson *et al.* [8] indicate that certain positive skewed and heavy-tailed distributions can model data center switch traffic, and highlights a difference between the data center environment and the wide area network, where the long-tailed Pareto distribution typically shows the best fit [8].

Mahalanobis Distance (MD) is a generalized distance which is useful for determining the similarity between an unknown sample and a collection of known samples, by considering the covariance between the variables and their mean values. The MD is a measure of the distance between a vector $\boldsymbol{x}$ and a distribution $\boldsymbol{X}$, introduced by P. C. Mahalanobis in 1936 [34]. MD is a multi-dimensional generalization for measuring how many standard deviations away $\boldsymbol{x}$ is from the mean $\boldsymbol{\mu}$ and covariance $\hat{\boldsymbol{\Sigma}}$ of $\boldsymbol{X}$. MD has been used for distance based anomaly detection with robust estimates in many areas, assuming that the MD between robust estimates and contaminated observations can reveal anomalies.

We propose a framework based on the distances between moments computed from learned robust subspace and contaminated observations, for anomaly detection on skewed and imbalanced data, and for botnet attack detection.

The proposed semi-supervised approach relies on learning robust subspace

from supposed legitimate traffic for estimating the moments (mean, skewness and kurtosis), and the anomaly detection is computed from contaminated observations by means of the distance between the robust moments and contaminated observations, without the computational cost of new robust subspace learning to detect anomalies on new observations. The m-RPCA can also be adopted as an unsupervised approach, where the robust subspace is learned from the contaminated data and the distance analysis is computed between the robust moments and the contaminated observations.

3. Data Model

In this paper, scalars are denoted by italic letters $(a, b, A, B, \alpha, \beta)$, vectors by lowercase bold letters $(\boldsymbol{a}, \boldsymbol{b})$, matrices by uppercase bold letters $(\boldsymbol{A}, \boldsymbol{B})$, and $a_{i,j}$ denotes the $(i,\ j)$ elements of the matrix $\boldsymbol{A}$. The superscripts $^{\mathrm{T}}$ and $^{-1}$ are used for matrix transposition and matrix inversion, respectively. We define that the Frobenius norm is denoted as $\|\cdot\|_F$, while $\|\cdot\|_*$ denotes the nuclear norm of a matrix and $\|\cdot\|_1$ means the sum of the absolute values of matrix entries. We also define the operator $\langle\cdot\rangle$, that is the standard trace inner product, and the operator $[\cdot]^c$, which denotes the indexes of the c largest values of a vector.

This section also presents a description of the simulated data model in Subsection 3.1 and in Subsection 3.2 we describe the CTU-13 data set.

3.1. The Simulated data set

We create two simulated data sets characterized by skewed and heavy tailed distributions, and create a simulated Gaussian distributed data set, to also analyze the detection rate of m-RPCA for not skewed and not heavy tailed distribution.

We selected Pareto (with scale $a = 3$, mode $m = 1$ and $\mu = 1$) and Log-normal (with mean $\mu = 0$ and standard deviation $\sigma = 1$) distributions, denoted respectively as $\boldsymbol{Y}_p$, while $\boldsymbol{Y}_l$, to simulate legitimate observations and to evaluate the anomaly detection on skewed data, by m-RPCA in comparison to widely

adopted algorithms for outlier detection. The simulated anomalies for Pareto and Lognormal distributions are white Gaussian noise with mean $\mu = 0$ and unitary standard deviation $\sigma = 1$. Note that Pareto and Lognormal distributions are skewed and heavy-tailed, and have been adopted to characterize network traffic of the Internet and data centers [8, 9].

We adopt the Gaussian distributed data $\boldsymbol{Y}_g$ to simulate legitimate observations, with mean $\mu = 0$ and unitary standard deviation $\sigma = 1$. An uniform noise between -6 and 6 is used to add anomalies into the Gaussian distributed data, where $\boldsymbol{Y}_g^c$ denotes the Gaussian data contaminated by an uniform distribution.

The Figure 1 shows a visual example of $\boldsymbol{Y}_p^c$ distribution, where the Pareto distribution $\boldsymbol{Y}_p$ is adopted for the majority class, and the Gaussian distribution simulate anomalies in comparison to the legitimate distribution, composing the Pareto distribution contaminated by Gaussian noise $\boldsymbol{Y}_p^c$.

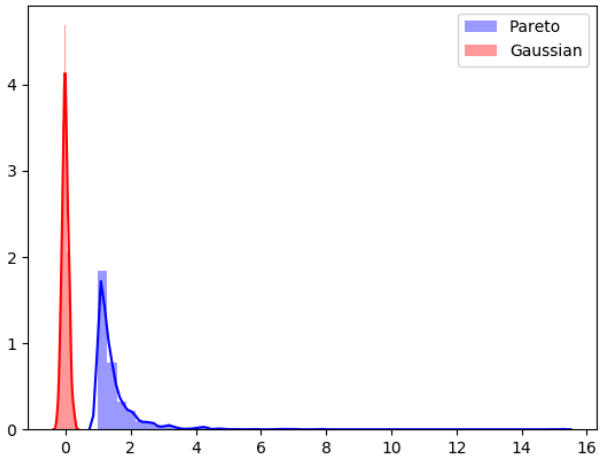
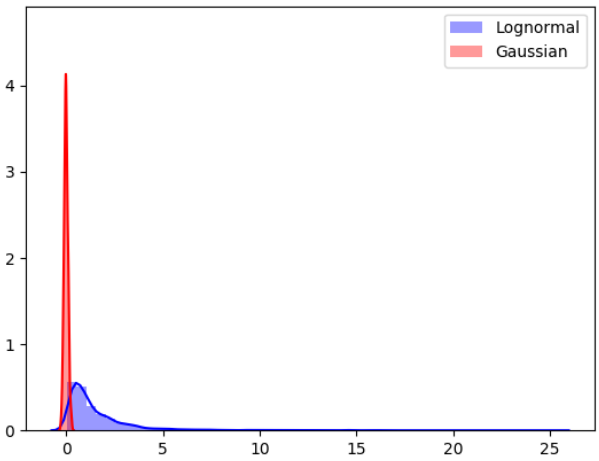


Figure 1: Pareto distribution and Gaussian contamination

The $\boldsymbol{Y}_l^c$ denotes the Lognormal distribution contaminated by Gaussian noise, accordingly to depicted by to Figure 2, where it is possible to observe the Lognormal legitimate distribution $\boldsymbol{Y}_l$ and the addictive gaussian noise to simulate the anomalies.Z



=

Figure 2: Lognormal distribution and Gaussian contamination

The figure 3 shows an example of the Gaussian distribution $\boldsymbol{Y}_g$ of legitimate observations and the uniform distribution to simulate the addition of anomalies, composing the Gaussian distribution contaminated by uniform noise $\boldsymbol{Y}_g^c$.

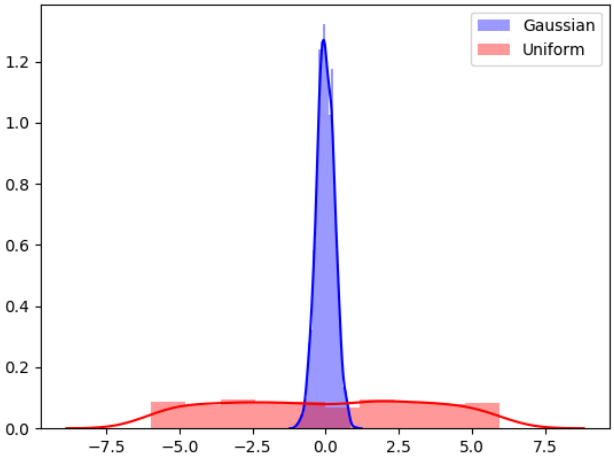


Figure 3: Example of Gaussian and Uniform Anomalies

Each contaminated data set is composed by a number of legitimate observations and contaminated samples. We evaluate contamination rates c between 1% and 50%, to simulate the imbalanced data of anomaly detection problems during our experiments. This data set simulate a total of 2400 events for each scenario with contaminated Pareto, Lognormal or Gaussian distribution. Therefore, the number of legitimate observations is defined according to the contamination rate selected for each evaluation.

3.2. The CTU-13 data set

The CTU-13 [15] is a data set of botnet traffic that was captured in the Czech Technical University, by means of a testbed and malware execution in a real network. The CTU-13 data set contains 13 scenarios with network flows of botnet malwares, that are: neris, rbot, virut, menti, sogou, nsys.ay and murlo. The botnet traffic is also classified as attack or command and control (C&C), while the legitimate flows can also be classified as legitimate or background.

The types of C&C and attack flows present in CTU-13 data set are:

- **Attacks:** Click Fraud (CF), Port Scan (PS), Fast Flux (FF), SPAM and DDoS;
- **C&C:** IRC, P2P and HTTP.

We refer to Garcia [35] and Garcia *et al.* [15] for a detailed description of the performed attacks and C&C flows, as well as for more information about the topology of the adopted testbed, rules for classifying legitimate flows, and an analysis of behaviors or patterns of the malware's traffic.

For all the scenarios, the authors of the CTU-13 data set convert the captured pcap files to NetFlows and release the processed flows. The data set provides ground-truth labels for flows as follows: flows from or to the infected machines are labeled as "botnet"; flows from or to well-known and controlled machines are labeled as "normal"; all other flows are labeled as "background."

Table 1 presents an overview grouped by scenario, according to the column ID, and shows the malwares used for botnet attacks, the types of attacks or

C&C, the total number of flows, the number of malicious flows which includes flows of C&C and attacks, and finally shows the number of legitimate flows.

Table 1: CTU-13 data set description

ID	Bot	Type	Total	Malicious	C&C	Attack	Normal
10	neris	IRC, Spam, CF	2,824,636	40,961 (1.45%)	341 (0.01%)	40,620 (1.44%)	30,387 (1.07%)
11	neris	IRC, Spam, CF	1,808,122	20,941 (1.16%)	673 (0.04%)	20,268 (1.12%)	9,120 (0.5%)
12	rbot	IRC, PS	4,710,638	26,822 (0.57%)	63 (0.00%)	26,759 (0.57%)	116,887 (2.48%)
15	rbot	IRC, DDoS	1,121,076	2,580 (0.23%)	52 (0.00%)	2,528 (0.23%)	25,268 (2.25%)
15-2	virut	Spam, PS, HTTP	129,832	901 (0.69%)	24 (0.02%)	877 (0.68%)	4,679 (3.6%)
16	menti	PS	558,919	4,630 (0.83%)	199 (0.04%)	4,431 (0.79%)	7,494 (1.34%)
16-2	sogou	HTTP	114,077	63 (0.06%)	26 (0.02%)	37 (0.03%)	1,677 (1.47%)
16-3	murlo	PS	2,954,230	6,127 (0.21%)	1,074 (0.04%)	5,053 (0.17%)	72,822 (2.46%)
17	neris	IRC, Spam, CF, PS	2,087,508	184,987 (8.86%)	2,973 (0.14%)	182,014 (8.72%)	43,340 (1.57%)
18	rbot	IRC, DDoS	1,309,791	106,352 (8.12%)	33 (0.00%)	106,319 (8.12%)	15,847 (1.2%)
18-2	rbot	IRC, DDoS	107,251	8,164 (7.61%)	2 (0.00%)	8,162 (7.61%)	2,718 (2.53%)
19	nsys.ay	P2P	325,471	2,168 (0.67%)	25 (0.01%)	2,143 (0.66%)	7,628 (2.35%)
15-3	virut	Spam, PS, HTTP	1,925,149	40,003 (2.08%)	536 (0.03%)	39,467 (2.05%)	31,939 (1.65)

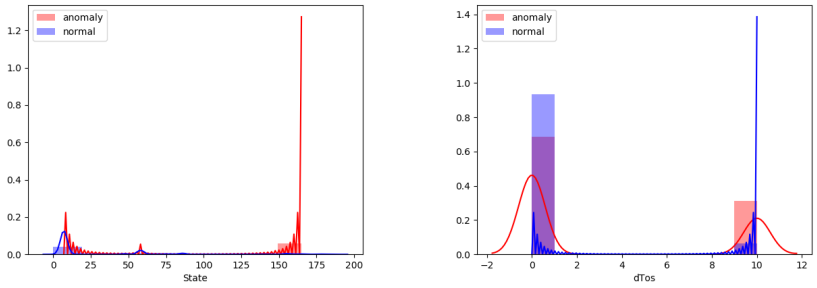
The full data set and scenarios can be denoted as $\boldsymbol{X} = \{\boldsymbol{X}_{10}, \boldsymbol{X}_{11}, \ldots, \boldsymbol{X}_{18-2}, \boldsymbol{X}_{19}\}$, in accordance to IDs presented in Table 1. In our experiment each contaminated scenario $\boldsymbol{X}_i$ is split into $\boldsymbol{X}_i^s$ containing 50% of the legitimate data, and into $\boldsymbol{X}_i^c$ that is composed by all anomalous flows and the necessary number of legitimate flows to have a testing data with the desired contamination rate.

The CTU-13 data set originally contains the following features for each flow: Start Time, End Time, Duration, Source IP Address, Source Port. Direction, Destination IP Address, Destination Port, State of TCP flags, Destination Type of Service, Source Type of Service, Total number of Packets, Total number of Bytes.

Our analysis of the available features leads to discard some features, considering that highly correlated features can bias or not improve the model, and that source or destination IP addresses can insert some false bias into learning models, since they can be changed by IP spoofing. Other risk related to adopt IP address for training models is the training model to learn that one IP is legitimate and this IP be infected subsequently, which can result into false negative

classifications.

We conducted and Exploratory Data Analysis (EDA) on the CTU-13 and observed that some features are skewed and present high overlapping between legitimate and anomalous flows, as can be seen in Figure 4(a) and Figure 4(b), that present the distributions of TCP state of scenario 16 and the type of services from destination of scenario 10.

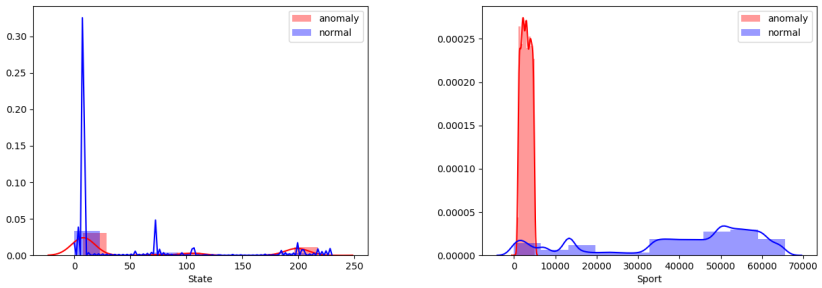


(a) State of scenario 16.

(b) Destination Type of service of scenario 10.

Figure 4: Example of skewness and overlapping

However, it is not possible to observe a pattern on distributions of all the features and scenarios of CTU-13, as depicted by Figures 5(a) and 5(b), that show the distributions of TCP states of the scenario 10 and source ports of scenario 16, and highlight the distributions of legitimate and anomalous flows.



(a) State of scenario 10.

(b) Source Port of scenario 16.

Figure 5: Skewness and Overlapping

Due to the number of available features and the class overlapping between legitimate and anomalous flows, we performed an correlation analysis and an empirical cross validation to identify the best set of features for network attack detection. Therefore, we adopt the following features: state, destination type of service, destination port, source port, total number of packets, total number of bytes and number of bytes from the source.

4. Moment Distances from Robust Subspace for Botnet Detection

This section describes the proposed approach for network attack detection by means of a distance analysis between moments computed from a robust subspace and contaminated observations of network traffic.

Robust subspace learning can be defined as the decomposition of a given data matrix $\boldsymbol{X} \in \mathbb{R}^{M \times N}$, with rows representing observations and columns representing features. In this work we adopt a flow based analysis, where $\boldsymbol{X}$ is modeled as a matrix of flows by features, and $\boldsymbol{X}$ is decomposed into the sum of a low rank matrix $\boldsymbol{L} \in \mathbb{R}^{M \times N}$, whose column subspace gives the robust principal components without outliers and noise, and a sparse matrix $\boldsymbol{S} \in \mathbb{R}^{M \times N}$, with element-wise outliers or noise.

Even though robust subspace learning has been adopted for network anomaly detection by means of highlights from the matrix $\boldsymbol{S}$, it was shown that $\boldsymbol{S}$ can indicate noise and outliers that can not be classified as malicious [30, 31], resulting into false positive classifications and requiring complementary approaches to obtain precise network anomaly detection [14].

Therefore, for network anomaly detection, we propose to learn a robust subspace from the legitimate traffic $\boldsymbol{X}^s$ for computing $\boldsymbol{L}^s$ and $\boldsymbol{S}^s$, followed by computing the robust moments, i.e. the mean $\boldsymbol{\mu}$, skewness $\boldsymbol{\epsilon}$ and kurtosis $\boldsymbol{\kappa}$, to evaluate the distance $\boldsymbol{d}$ between contaminated observations $\boldsymbol{X}^c$ and robust moments. The largest distances are classified as anomalous and indicate network attacks, denoted as $\boldsymbol{N}$ by the data model $\boldsymbol{X} = \boldsymbol{U} + \boldsymbol{N}$, where $\boldsymbol{U}$ denotes the legitimate network traffic.

RPCA is a well-known method to recover a low-rank matrix $\boldsymbol{L}$ and sparse matrix $\boldsymbol{S}$ from corrupted measurements modeled as $\boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S}$. This decomposition in low-rank and sparse matrices can be achieved by techniques such as Principal Component Pursuit method (PCP), and by optimization methods, such as the Augmented Lagrange Multiplier Method (ALM), Alternating Direction Method (ADM), Fast Alternating Minimization (FAM) or Iteratively Reweighted Least Squares (IRLS) [29, 30, 31].

According to Wright et al. [36], under rather broad conditions, as long as the error matrix $\boldsymbol{S}$ is sufficiently sparse, it is possible to recover a low-rank matrix by solving the following convex optimization problem:

$$(\hat{\boldsymbol{L}}, \hat{\boldsymbol{S}}) \leftarrow \min_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{S}\|_1 \tag{1}$$

$$\text{subject to: } \boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S}$$

$$\|\boldsymbol{L}\|_* = \sum_i \sigma_i(\boldsymbol{L}) \tag{2}$$

$$\|\boldsymbol{S}\|_1 = \sum_{ij} |\boldsymbol{S}_{ij}| \tag{3}$$

where $\|\cdot\|_*$ denotes the nuclear norm of a matrix, λ is a positive weighting parameter, which determines the sparsity of $\boldsymbol{S}$, and $\|\cdot\|_1$ means the sum of the absolute values of matrix entries, and σ denotes the singular values of a matrix.

Before ALM, some methods were proposed to solve that convex optimization problem, such as Iterative Thresholding (IT) and Accelerated Proximal Gradient (APG). However, according to Zhouchen *et al.* [37], both approaches have scalability problems and require a large number of iterations to converge. The Augmented Lagrange Multiplier (ALM) is proven to have a *Q-linear* convergence speed and experimental results show that ALM is five times faster than APG, which in theory is sub-linear [37]. Furthermore, ALM reaches more accurate results with less iterations.

The RPCA with ALM can be formulated as:

$$l(\boldsymbol{L},\boldsymbol{S},\boldsymbol{Y}) = \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{S}\|_1 + \langle \boldsymbol{Y}, \boldsymbol{X}-\boldsymbol{L}-\boldsymbol{S}\rangle + \frac{\mu}{2}\|\boldsymbol{X}-\boldsymbol{L}-\boldsymbol{S}\|_F^2, \quad (4)$$

where $\boldsymbol{Y}$ is the multiplier of the linear constraint and μ is the penalty parameter for the violation of the linear constraint [38].

Thus, an iterative scheme can be presented as:

$$\begin{cases} (\boldsymbol{L}_{k+1}, \boldsymbol{S}_{k+1}) \in_{\boldsymbol{L},\boldsymbol{S}\in\mathbb{R}^{m\times n}} \{l(\boldsymbol{L},\boldsymbol{S},\boldsymbol{Y}_k)\}, \\ \boldsymbol{Y}_{k+1} = \boldsymbol{Y}_k + \mu(\boldsymbol{X}-\boldsymbol{L}_k-\boldsymbol{S}_k), \end{cases} \quad (5)$$

We adopt RPCA with ADM, which, generally speaking, is a practical improvement of the classical ALM method for solving convex programming problem with linear constraints, by fully taking advantage of its high-level separable structure [38]. ADM minimizes $\boldsymbol{L}$ and $\boldsymbol{S}$ variables serially by solving the following problems to generate the new iterate:

$$\begin{cases} \boldsymbol{L}_{k+1} \in_{\boldsymbol{L}\in\mathbb{R}^{m\times n}} \{l(\boldsymbol{L},\boldsymbol{S}_k,\boldsymbol{Y}_k)\} \\ \boldsymbol{S}_{k+1} \in_{\boldsymbol{S}\in\mathbb{R}^{m\times n}} \{l(\boldsymbol{L}_{k+1},\boldsymbol{S},\boldsymbol{Y}_k)\} \\ \boldsymbol{Y}_{k+1} = \boldsymbol{Y}_k + \mu(\boldsymbol{X}-\boldsymbol{L}_k-\boldsymbol{S}_k) \end{cases} \quad (6)$$

Moments are a set of statistical parameters to measure a distribution. The arithmetic mean is the first general moment, the second is the variance, while skewness (asymmetry) is the third moment and kurtosis (tailedness) is the fourth moment [39].

Let the mean $\boldsymbol{\mu} \in \mathbb{R}^{1\times N}$ be

$$\boldsymbol{\mu} = \frac{1}{M}\sum_{i=1}^{M} \boldsymbol{x}_i, \quad (7)$$

where M is the number of samples, and let the sample covariance matrix

$\hat{\boldsymbol{\Sigma}} \in \mathbb{R}^{N \times N}$ be

$$\hat{\boldsymbol{\Sigma}} = \frac{1}{N-1} \sum_{i=1} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T, \tag{8}$$

According to Zwillinger and Kokoska [40], the general expression for the p-th moment $\boldsymbol{m}_p \in \mathbb{R}^{1 \times N}$ about the mean $\boldsymbol{\mu}$ is given by

$$\boldsymbol{m}_p = \frac{1}{M} \sum_{i=1}^{M} (\boldsymbol{x}_i - \boldsymbol{\mu})^p. \tag{9}$$

Therefore, Zwillinger and Kokoska [40] present that the skewness $\boldsymbol{\epsilon} \in \mathbb{R}^{1 \times N}$ about the mean $\boldsymbol{\mu}$ is calculated by

$$\boldsymbol{\epsilon} = \frac{\boldsymbol{m}_3}{\boldsymbol{m}_2^{\frac{3}{2}}}, \tag{10}$$

and the kurtosis $\boldsymbol{\kappa} \in \mathbb{R}^{1 \times N}$ is given as

$$\boldsymbol{\kappa} = \frac{\boldsymbol{m}_4}{\boldsymbol{m}_2^2}. \tag{11}$$

We propose to compute the mean $\boldsymbol{\mu}$, the skewness $\boldsymbol{\epsilon}$, the kurtosis $\boldsymbol{\kappa}$ and the covariance matrix $\hat{\boldsymbol{\Sigma}}$ from the robust subspace $\boldsymbol{L}^s$, after to minimize the (6) for $\boldsymbol{X}^s$. We also propose to compute the Mahalanobis Distance (MD) for detecting anomalies between contaminated observations and the moments calculated from a robust subspace computed by RPCA.

The classical Mahalanobis Distance is defined as

$$\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\mu}, \hat{\boldsymbol{\Sigma}}) = \sqrt{(\boldsymbol{x} - \boldsymbol{\mu})\hat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})^T}, \tag{12}$$

where $\boldsymbol{x}$ is a vector of a new observations, $\boldsymbol{\mu}$ is the mean vector of known observations, also referred as location, and $\hat{\boldsymbol{\Sigma}}$ is the covariance matrix of known observations, also referred as scatter. The classical MD usually relies on robust mean and robust covariance matrix for outlier detection, which are commonly

computed by MCD [41, 42]. Here we propose to compute the Robust-Mean Distance $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\mu}, \hat{\boldsymbol{\Sigma}})$ according to (12), but adopting the mean $\boldsymbol{\mu}$ and covariance matrix $\hat{\boldsymbol{\Sigma}}$ calculated from a robust subspace $\boldsymbol{L}^s$ learned by RPCA.

We also propose to extend (12) to implement the Robust-Skewness Distance $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\epsilon}, \hat{\boldsymbol{\Sigma}})$, as follows:

$$\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\epsilon}, \hat{\boldsymbol{\Sigma}}) = \sqrt{(\boldsymbol{x} - \boldsymbol{\epsilon})\hat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x} - \boldsymbol{\epsilon})^T}. \tag{13}$$

Finally, we propose to extend (12) to implement the Robust-Kurtosis Distance $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\kappa}, \hat{\boldsymbol{\Sigma}})$, as follows:

$$\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\kappa}, \hat{\boldsymbol{\Sigma}}) = \sqrt{(\boldsymbol{x} - \boldsymbol{\kappa})\hat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x} - \boldsymbol{\kappa})^T}. \tag{14}$$

The distances $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\mu}, \hat{\boldsymbol{\Sigma}})$, $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\epsilon}, \hat{\boldsymbol{\Sigma}})$ and $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\kappa}, \hat{\boldsymbol{\Sigma}})$ shall be computed and evaluated separately and independently, for network attack detection based on robust mean, robust skewness and robust kurtosis, respectively.

Therefore, the robust subspace $\boldsymbol{L}^s$ learned from legitimate data $\boldsymbol{X}^s$ followed by the Robust-Mean Distance $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\mu}, \hat{\boldsymbol{\Sigma}})$ is called as Mean Distance of Robust Principal Component Analysis (md-RPCA), or is called Skewness Distance of Robust Principal Component Analysis (sd-RPCA) when followed by Robust-Skewness Distance, given by $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\epsilon}, \hat{\boldsymbol{\Sigma}})$, or Kurtosis Distance of Robust Principal Component Analysis (kd-RPCA) when followed by Robust-Kurtosis Distance, given by $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\kappa}, \hat{\boldsymbol{\Sigma}})$.

The contamination rate, denoted by c, is a parameter traditionally adopted by well established outlier detection algorithms [43], and refers to the percentage rate of observations that are known as anomalous. The contamination rate can be well-known for some areas, or can be computed by cross-validation [44] or can be assumed according to previous observations. In our proposal, the contamination defines the number of the largest distances $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\mu}, \hat{\boldsymbol{\Sigma}})$, $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\epsilon}, \hat{\boldsymbol{\Sigma}})$ or $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\kappa}, \hat{\boldsymbol{\Sigma}})$ that shall be classified as anomalous. The observations classified as legitimate and anomalous, according to the contamination c, are denoted by the vector $\hat{\boldsymbol{t}}$, with values of 1 for observations classified as anomalous and 0 to

denote legitimate observations.

The Algorithm 1 describes all possible steps of m-RPCA and the approaches md-RPCA, sd-RPCA and kd-RPCA, for the semi-supervised learning approach. The unsupervised approach adopts the same steps, but adopting the contaminated data for robust subspace learning, and requiring new robust subspace learning for testing anomaly detection on new set of observations.

Algorithm 1: Moment Distances from Robust Subspace

Result: $\hat{\boldsymbol{t}}_{\mu}, \hat{\boldsymbol{t}}_{\epsilon}, \hat{\boldsymbol{t}}_{\kappa}$

1 Given $\boldsymbol{X}$ split into $\boldsymbol{X}^s$ and $\boldsymbol{X}^c$;

2 **while** not $\min_{L,S} \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{S}\|_1$ from $\boldsymbol{X}^s$ **do**

3 $\quad \boldsymbol{L}_{k+1} \in_{\boldsymbol{L} \in \mathbb{R}^{m \times n}} \{l(\boldsymbol{L}, \boldsymbol{S}_k, \boldsymbol{Y}_k)\}$;

4 $\quad \boldsymbol{S}_{k+1} \in_{\boldsymbol{S} \in \mathbb{R}^{m \times n}} \{l(\boldsymbol{L}_{k+1}, \boldsymbol{S}, \boldsymbol{Y}_k)\}$;

5 $\quad \boldsymbol{Y}_{k+1} = \boldsymbol{Y}_k + \mu(\boldsymbol{X} - \boldsymbol{L}_k - \boldsymbol{S}_k)$;

6 **end**

7 $\boldsymbol{\mu} = \dfrac{1}{M} \sum_{i=1}^{M} \boldsymbol{x}_i$, from $\boldsymbol{L}$;

8 $\hat{\boldsymbol{\Sigma}} = \dfrac{1}{N-1} \sum_{i=1}^{N} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T$, from $\boldsymbol{L}$;

9 $\boldsymbol{\epsilon} = \dfrac{\boldsymbol{m}_3}{\boldsymbol{m}_2^{\frac{3}{2}}}$, from $\boldsymbol{L}$;

10 $\boldsymbol{\kappa} = \dfrac{\boldsymbol{m}_4}{\boldsymbol{m}_2^2}$, from $\boldsymbol{L}$;

11 $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\mu}, \hat{\boldsymbol{\Sigma}}) = \sqrt{(\boldsymbol{x} - \boldsymbol{\mu})\hat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})^T}$, from $\boldsymbol{X}^c$;

12 $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\epsilon}, \hat{\boldsymbol{\Sigma}}) = \sqrt{(\boldsymbol{x} - \boldsymbol{\epsilon})\hat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x} - \boldsymbol{\epsilon})^T}$, from $\boldsymbol{X}^c$;

13 $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\kappa}, \hat{\boldsymbol{\Sigma}}) = \sqrt{(\boldsymbol{x} - \boldsymbol{\kappa})\hat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{x} - \boldsymbol{\kappa})^T}$, from $\boldsymbol{X}^c$;

14 $\hat{\boldsymbol{t}}_{\mu} = [\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\mu}, \hat{\boldsymbol{\Sigma}})]^c$;

15 $\hat{\boldsymbol{t}}_{\epsilon} = [\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\epsilon}, \hat{\boldsymbol{\Sigma}})]^c$;

16 $\hat{\boldsymbol{t}}_{\kappa} = [\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{\kappa}, \hat{\boldsymbol{\Sigma}})]^c$;

The steps between 1 and 10 of the Algorithm 1 are the training from legitimate data $\boldsymbol{X}^s$, which are common steps shared by md-RPCA, sd-RPCA and kd-RPCA. The steps between 11 and 16 aim the anomaly detection from new contaminated observations, by means of Mahalanobis Distance of robust moments. Note that the steps 11 and 14 refer to the steps of md-RPCA for anomaly detection, while the steps 12 and 15 refer to the sd-RPCA, and the

steps 13 and 16 refer to the steps of kd-RPCA. It is important to highlight that the anomaly detection from new observations does not require new robust subspace learning when adopting the semi-supervised approach, which only requires to compute a moment-based distance to classify the c largest distances as anomalous or legitimate.

The m-RPCA can also be adopted as unsupervised algorithm if the lines 1 and 2 of the Algorithm 1 are changed to substitute $\boldsymbol{X}^s$ by $\boldsymbol{X}^c$. Hence, the robust subspace is learned from contaminated data and used for comparing the distance between moments from robust estimate and contaminated observations. It is important to note that this unsupervised approach requires the computational cost of computing new subspace learning for any new set of observations.

5. Experiments and Results

This section presents the performed experiments on simulated and real data set for anomaly detection. First, in Section 5.1 we present the adopted metric to evaluate imbalanced data in the context of anomaly detection. In Section 5.2 we describe the experiment for anomaly detection on simulated skewed and heavy-tailed data set, and we present in Section 3.2 the experiment for network anomaly detection on CTU-13 data set.

5.1. The metric

In anomalous detection problems, where anomalies are rare events, if one classify all observations as legitimate and apply an accuracy evaluation, one would have high accuracy but poor true-positive detection. Due to the importance given by the F_1 (also referred as F-score or F-measure) to true-positive detection in scenarios such as the network attack detection, it is the preferable measure for imbalanced data [45, 6]. Therefore, F_1 is the metric used for validation of our experiments.

The F_1 is the harmonic mean of precision and recall, where p_p denotes the

precision and recall is denoted as r_c. The F_1 is given by

$$F_1 = 2 \cdot \frac{p_r \cdot r_c}{p_r + r_c}. \tag{15}$$

Precision can be seen as a measure of exactness or accuracy, that relies on true positive and false positive measures, denoted by t_p and f_p, respectively. The precision is defined by

$$p_r = \frac{t_p}{t_p + f_p}. \tag{16}$$

Recall is a measure of completeness, to calculate proportion of actual positives was identified correctly, by means of the true positive and false negative measures, denoted by t_p and f_n. The recall is defined by

$$r_c = \frac{t_p}{t_p + f_n}. \tag{17}$$

This experimental evaluation compares our proposal to widely adopted algorithms for anomaly detection that also rely on contamination rate for anomaly detection, which are: A PCA approach based on the sum of weighted projected distances to the eigenvector hyperplanes [46]; MCD [41, 42]; One-Class Support Vector Machines [47]; Local Outlier Factor (LOF) [48]; k-Nearest Neighbors [49]; and Isolation Forest [50].

We also compare the results of our proposals to ROBPCA-AO for anomaly detection on simulated and CTU-13 data set, considering that ROBPCA-AO also relies on robust estimates with adjusted outlyingness based on robust skewness [12].

5.2. Simulated Experiment

Anomaly detection algorithms usually rely on supervised or unsupervised methods, where the former requires labeled legitimate and anomalous data for training anomaly detection models, while the latter does not require labeled data or training. Semi-supervised algorithms are an alternative for the anomaly detection problem, considering that this method only relies on legitimate data

for training and that non-malicious data can be obtained from historical information and from rule-based approaches.

We propose semi-supervised and unsupervised approaches for m-RPCA, where the former relies on legitimate data $\boldsymbol{X}^s$ for training and on contamination rate c for anomaly detection, while the latter relies only on contaminated data $\boldsymbol{X}^c$ for robust subspace learning and relies on contamination rate c to select the largest distances. We assume that c is well-known or can be estimated for real world problems of anomaly detection, in accordance to the assumption adopted by the well established algorithms [43] selected for comparison, that also rely on contamination rate $\boldsymbol{X}^c$.

The availability of labeled data is a challenging concern in real world problems of anomalous detection, where anomalies are rare or even unknown events. Considering that RPCA has already been adopted to isolate outliers from training data [14], we also propose to evaluate the m-RPCA for a contaminated semi-supervised approach based on training from contaminated data, to evaluate the impact that contaminated data for robust subspace learning can cause in the anomaly detection results.

Therefore, we propose to evaluate the following approaches for m-RPCA: semi-supervised, contaminated semi-supervised, and unsupervised.

For the semi-supervised approach we propose to learn the robust subspace and compute the moments from the legitimate data $\boldsymbol{Y}_g$, $\boldsymbol{Y}_p$ and $\boldsymbol{Y}_l$ with Gaussian, Pareto and Lognormal distributions, respectively, and test the anomaly detection for contaminated data $\boldsymbol{Y}_g^c$, $\boldsymbol{Y}_p^c$ and $\boldsymbol{Y}_l^c$.

For the Contaminated Semi-supervised approach we evaluate the robustness of the m-RPCA approaches for learning from contaminated training data, to analyze if m-RPCA can be an alternative for the lack of known legitimate data. Therefore we propose to train the model from a contaminated legitimate data $\boldsymbol{Y}_g^{c'}$, $\boldsymbol{Y}_p^{c'}$ and $\boldsymbol{Y}_l^{c'}$, with the same contamination rate of the testing data, but without data repetition between training and testing, taking into account that we shall consider a different contaminated data but adopt the same contamination rate for training and testing.

We finally evaluate the unsupervised approach, that relies on the test data $\boldsymbol{Y}_g^c$, $\boldsymbol{Y}_p^c$ and $\boldsymbol{Y}_l^c$ for robust subspace learning, and then classify the results according to the distance between the contaminated data $\boldsymbol{Y}_g^c$, $\boldsymbol{Y}_p^c$ and $\boldsymbol{Y}_l^c$, and the moments computed from the learned robust subspace.

5.3. Experiment for CTU-13

In contrast to Garcia *et al.* [15], we propose to evaluate each scenario of the CTU-13 data set individually, in a semi-supervised approach that does not rely on training data with labeled anomalies, to evaluate our proposed approaches to all botnet malwares of the CTU-13 data set.

In our experiment setup each contaminated scenario $\boldsymbol{X}_i$ of CTU-13 is split into $\boldsymbol{X}_i^s$, containing 50% of the legitimate data, and into test data $\boldsymbol{X}_i^c$ containing 33% of legitimate and 67% of anomalous data. We adopt a contamination rate c of 33% for experimenting anomaly detection for the CTU-13 data set, considering that this contamination rate presented good results on the simulated experiment for our proposals and for some selected algorithms.

This experiment consider only the semi-supervised approach due to results of simulated experiments on simulated data set presented in Subsection 5.4, that shows better results for the semi-supervised approach and highlight that the this approach can obtain good results even when trained with contaminated data set.

5.4. Results of Simulated Experiment

We adopt prefixes to denote the evaluated approaches for md-RPCA, sd-RPCA and kd-RPCA, which are ss_ to denote semi-supervised, css_ to denote contaminated semi-supervised and u_ for unsupervised.

Initially, we evaluated the F_1 of the selected algorithms and m-RPCA approaches for anomaly detection on Gaussian distributed legitimate data contaminated by uniform distributed anomalies. The Figure 6 shows the F_1 over the contamination rate between 1% and 50% for m-RPCA approaches and Isolation Forest (IF) [50], k-Nearest Neighbors (KNN) [49], Local Outlier Factor (LOF)

[48], Minimum Covariance Deteminant (MCD) [42], One-Class Support Vector Machines (OCSVM) [47], PCA [46] and ROBPCA-AO [12].

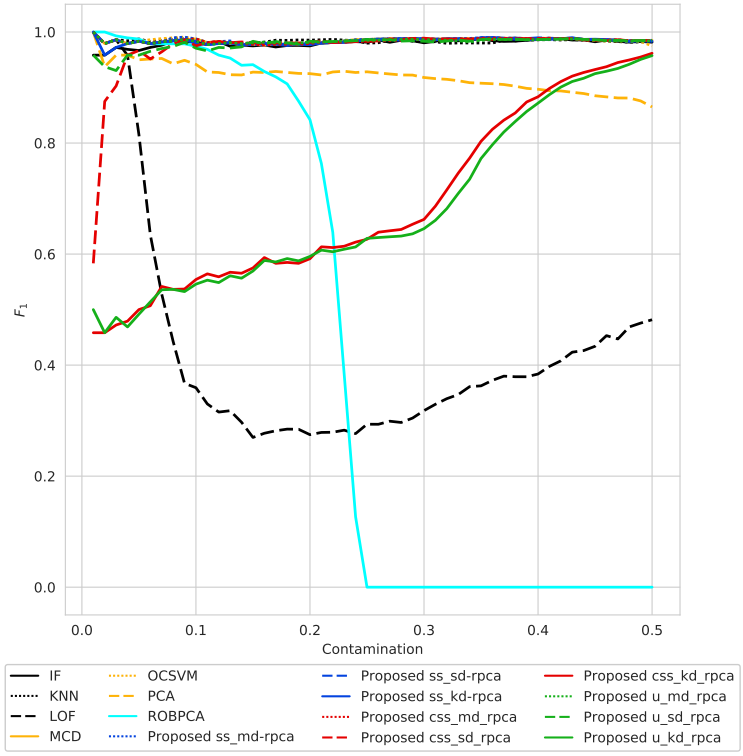


Figure 6: Anomaly detection on Gaussian distributed with uniform anomalies

It is possible to observe in Figure 6 that LOF, PCA, css_kd-RPCA and u_kd-RPCA presented lower performance than the remain algorithms, that obtain results higher than 0.95 in average, for anomaly detection on Gaussian distributed data. The exception is the ROBPCA-AO, that presented high score for lower contamination but decreased with the contamination increasing.

Note that the css_kd-RPCA and u_kd-RPCA are the contaminated semi-supervised and unsupervised versions of kd-RPCA, that obtain worse results than the semi-supervised approach of kd-RPCA, for anomaly detection on Gaussian distributed data contaminated by uniform anomalies. However, the unsupervised versions of md-RPCA and sd-RPCA presented similar results to widely

adopted unsupervised algorithms for outlier detection. The results also show that the semi-supervised approach of md-RPCA, sd-RPCA and kd-RPCA obtain high anomaly detection rate on Gaussian distributed data and presented similar results to the unsupervised algorithms IF, KNN, MCD and OCSVM.

The Figure 7 and 8 show the results for anomaly detection on skewed and heavy tailed distributions, presenting results for anomaly detection on Pareto and Lognormal distributions, respectively.

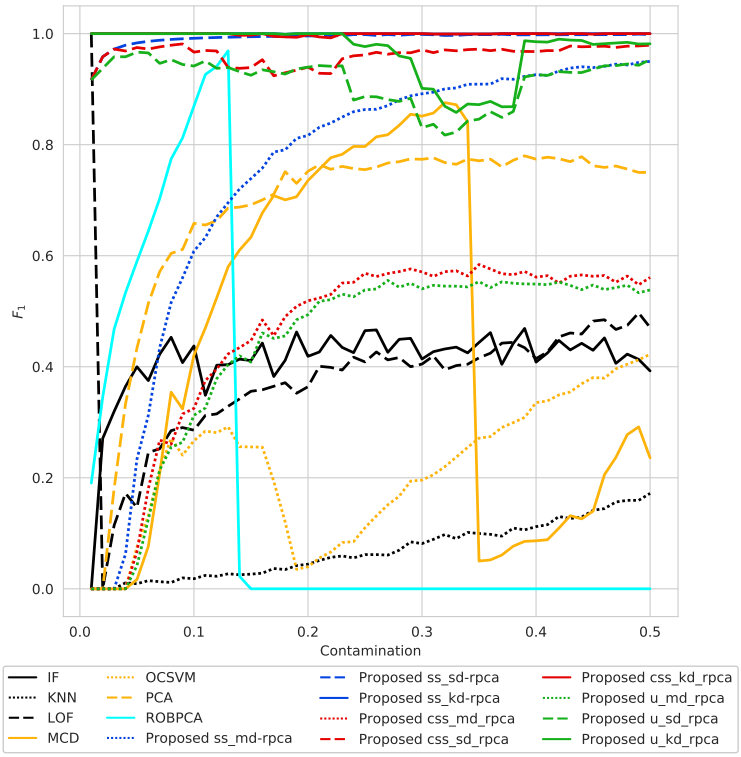


Figure 7: Anomaly detection on Pareto distributed with Gaussian anomalies

The Figure 7 depics the results for anomaly detection on Pareto legitimate data with Gaussian anomalies, and shows that IF, KNN, LOF, OCSVM, css_md-RPCA and u_md-RPCA performed worse than the remain algorithms for the evaluated contamination, with results lower than 0.6 even with higher contamination. Note that the approaches of m-RPCA with lower scores are

css_md-RPCA and u_md-RPCA, which are mean-based approaches. However, the ss_md-RPCA is the mean-based approach that presented lower results for lower contamination but achieved more than 0.8 with contamination near of 0.2 or higher, and achieved results better than MCD and PCA.

Figure 7 shows that ROBPCA-AO presented high F_1 score for low contamination rates, presenting better results than ss_md-RPCA, MCD and PCA for anomaly detection on Pareto distributed data, initially. However, the results of ROBPCA-AO decrease drastically with the contamination increasing. It is also possible to observe in Figure 7 that all approaches based on kd-RPCA initially achieved the best results for anomaly detection on Pareto distributed data, but the results for the unsupervised variate with contamination near of 0.2 or higher, while the ss_kd-RPCA presents stable results, near of 1.0 F_1, for all evaluated contamination rates.

All approaches based on sd-RPCA obtained a F_1 score higher than 0.8 for anomaly detection on Pareto distributed data, however the unsupervised and contaminated semi-supervised presented high variation of F_1 and lower results in comparison to the semi-supervised sd-RPCA. The contaminated semi-supervised approaches of sd-RPCA and kd-RPCA presented higher results than 0.8 and similar to the semi-supervised and unsupervised approaches of sd-RPCA and kd-RPCA. These results highlight the resilience of the robust subspace learning even for contaminated training data.

The unsupervised approaches of sd-RPCA and kd-RPCA presented more than 0.8 for all contamination rate, showing better results than the widely adopted unsupervised algorithms for outlier detection. However, u_kd-RPCA and u_sd-RPCA presented lower results than ss_kd-RPCA and ss_sd-RPCA.

Therefore, the semi-supervised approaches of m-RPCA overcome other approaches of m-RPCA and overcome the selected algorithms for anomaly detection on Pareto distributed data with Gaussian anomalies. Finally, it is possible to highlight the semi-supervised kd-RPCA, which obtained stable results near of 1.0 F_1 for all evaluated contamination.

The results for anomaly detection on Lognormal legitimate data with Gaus-

sian anomalies is depicted by Figure 8, and show that IF, KNN, LOF, MCD, OCSVM, css_md-RPCA, u_md-RPCA and ROBPCA-AO perform worse than the remain evaluated algorithms, with results lower than 0.6, even with higher contamination.

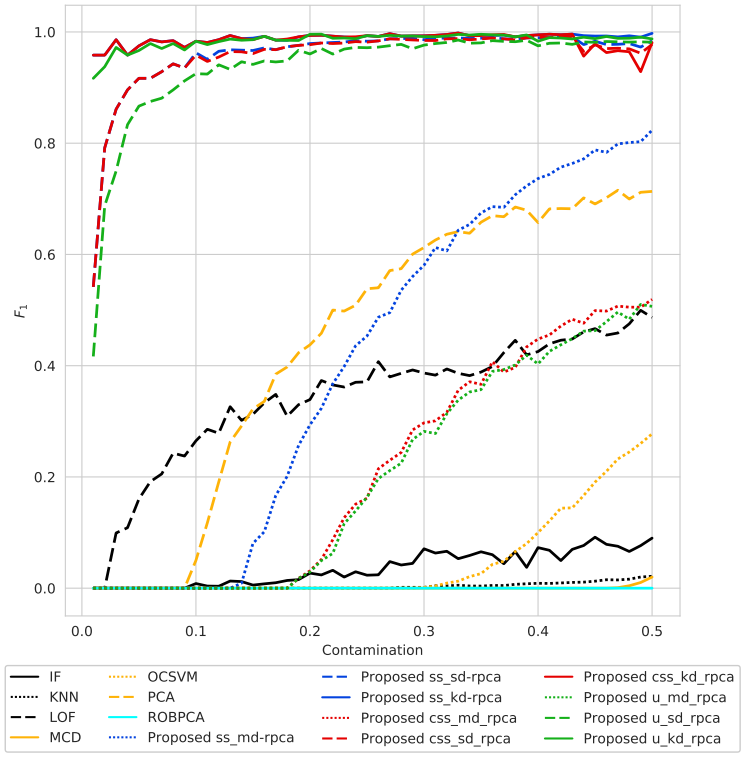


Figure 8: Anomaly detection on Lognormal distributed with Gaussian anomalies

The ss_md-RPCA and PCA algorithms perform similar for anomaly detection on Lognormal distributed data, with worse detection rate for lower contamination and better performance with contamination higher than 0.4. However, the results of ss_md-RPCA and PCA are lower than all approaches of sd-RPCA and kd-RPCA.

It is important to note that all mean-based approaches of m-RPCA presented lower results for anomaly detection on Lognormal data, in comparison to approaches based on skewness (sd-RPCA) and kurtosis (kd-RPCA), that achieved

the best anomaly detection rates. However, the approaches of md-RPCA presented better results than IF, KNN, MCD, OCSVM and ROBPCA-AO.

The approaches based on kd-RPCA presented higher detection rate for all contamination, with scores near of 1.0. The semi-supervised and contaminated semi-supervised approaches for sd-RPCA performed similarly, but the unsupervised approach of sd-RPCA presented lower anomaly detection on Lognormal distributed data, for lower contamination.

It is possible to note that the contaminated semi-supervised approaches of kd-RPCA and sd-RPCA performed similar to the semi-supervised approach of the same algorithms. These results highlight the resilience of the robust subspace learning, even from contaminated training data.

Finally, it is possible to observe that the semi-supervised approaches of m-RPCA overcome other approaches of m-RPCA and all the selected algorithms for anomaly detection on Lognormal distributed data with Gaussian anomalies.

Following we present the Tables 2, 3 and 4 to show the results of the selected algorithms and m-RPCA for anomaly detection on Gaussian, Pareto and Lognormal distributions, with 10%, 25% and 33% of contamination rate. The columns present the F_1 score for each scenario, sorted by the best F_1.

Table 2: Results for Simulated data set with 33% of contamination

Algorithm	**Gaussian +Uniform**	**Pareto+ Gaussian**	**Lognormal +Gaussian**
Proposed Semi-Supervised sd-RPCA	0.98	0.99	0.97
Proposed Semi-Supervised kd-RPCA	0.97	0.99	0.98
Proposed Contaminated Semi-Supervised sd-RPCA	0.98	0.94	0.96
Proposed Unsupervised sd-RPCA	0.98	0.82	0.93
Proposed Contaminated Semi-Supervised kd-RPCA	0.76	0.99	0.98
Proposed Unsupervised kd-RPCA	0.74	0.86	0.99
Proposed Semi-Supervised md-RPCA	0.98	0.90	0.66
PCA [46]	0.87	0.76	0.62
Proposed Contaminated Semi-Supervised md-RPCA	0.98	0.55	0.35
Proposed Unsupervised md-RPCA	0.98	0.54	0.31
Isolation Forest [50]	0.98	0.44	0.05
One-class SVM [47]	0.98	0.24	0.00
KNN [49]	0.98	0.08	0.00
MCD [42]	0.98	0.02	0.00
LOF [48]	0.35	0.40	0.36
ROBPCA-AO [12]	0.00	0.00	0.00

The results for 33% of contamination show high scores of anomaly detection for Gaussian data with uniform anomalies for the evaluated algorithms,

with exception of LOF and ROBPCA-AO. The semi-supervised approaches of m-RPCA presented the highest results for anomaly detection on Pareto data with Gaussian anomalies, and for anomaly detection on Lognormal data with Gaussian anomalies, as well as for Gaussian data with uniform anomalies.

Table 3: Results for Simulated data set with 25% of contamination

Algorithm	**Gaussian +Uniform**	**Pareto+ Gaussian**	**Lognormal +Gaussian**
Proposed Semi-Supervised kd-RPCA	0.96	1.00	0.98
Proposed Semi-Supervised sd-RPCA	0.97	0.99	0.96
Proposed Contaminated Semi-Supervised sd-RPCA	0.97	0.91	0.95
Proposed Unsupervised sd-RPCA	0.97	0.88	0.91
Proposed Contaminated Semi-Supervised kd-RPCA	0.63	1.00	0.97
Proposed Unsupervised kd-RPCA	0.62	0.97	0.99
Proposed Semi-Supervised md-RPCA	0.97	0.85	0.49
PCA [46]	0.91	0.75	0.56
MCD [42]	0.97	0.79	0.00
Proposed Contaminated Semi-Supervised md-RPCA	0.97	0.54	0.17
Proposed Unsupervised md-RPCA	0.97	0.53	0.14
Isolation Forest [50]	0.97	0.46	0.05
One-class SVM [47]	0.97	0.11	0.00
KNN [49]	0.97	0.05	0.00
LOF [48]	0.27	0.33	0.30
ROBPCA-AO [12]	0.00	0.00	0.00

The Table 3 also shows higher F_1 scores of semi-supervised m-RPCA for 25% of contamination, in comparison to the remain evaluated algorithms. It is important to note that the highest F_1 score for anomaly detection on Lognormal distributed data with Gaussian contamination was the unsupervised kd-RPCA, with 0.99 F_1 score, and note that the contaminated semi-supervised approaches of sd-RPCA and kd-RPCA presented F_1 scores near of the results for semi-supervised approaches in scenarios with Gaussian, Pareto and Lognormal legitimate data.

The results for contamination rate of 10% shown by Table 4 shows worse F_1 score in comparison to contamination of 25% and 33%. However, the ROBPCA-AO presented high scores for anomaly detection on Gaussian data, overcoming LOF, css_kd-RPCA and u_kd-RPCA. ROBPCA-AO also presented better results for anomaly detection on Pareto data in comparison to u_md-RPCA, css_md-RCAP, ss_md-RCAP, PCA, OCSVM, LOF, KNN and IF.

Therefore, it is possible to observe that the md-RPCA, sd-RPCA and kd-RPCA approaches, which are based on robust subspace learning, presented

Table 4: Results for Simulated data set with 10% of contamination

Algorithm	Gaussian +Uniform	Pareto+ Gaussian	Lognormal +Gaussian
Proposed Semi-Supervised kd-RPCA	0.97	1.00	0.95
Proposed Semi-Supervised sd-RPCA	0.97	0.98	0.84
Proposed Contaminated Semi-Supervised sd-RPCA	0.96	0.90	0.82
Proposed Unsupervised sd-RPCA	0.97	0.93	0.70
Proposed Unsupervised kd-RPCA	0.50	0.97	0.98
Proposed Contaminated Semi-Supervised kd-RPCA	0.48	1.00	0.94
ROBPCA-AO [12]	0.97	0.87	0.00
PCA [46]	0.92	0.65	0.06
Proposed Semi-Supervised md-RPCA	0.97	0.59	0.00
Isolation Forest [50]	0.97	0.43	0.00
MCD [42]	0.97	0.40	0.00
Proposed Unsupervised md-RPCA	0.97	0.31	0.00
Proposed Contaminated Semi-Supervised md-RPCA	0.97	0.28	0.00
One-class SVM [47]	0.97	0.26	0.00
KNN [49]	0.97	0.01	0.00
LOF [48]	0.29	0.27	0.24

higher anomaly detection from imbalanced and skewed data, in comparison to widely adopted algorithms for anomaly detection.

5.5. Results of CTU-13 Experiment

In this section we present the experiment on network anomaly detection from the CTU-13 data set, evaluating the results of md-RPCA, sd-RPCA, kd-RPCA, Isolation Forest (IF) [50], k-Nearest Neighbors (KNN) [49], Local Outlier Factor (LOF) [48], Minimum Covariance Deteminant (MCD) [42], One-Class Support Vector Machines (OCSVM) [47], PCA [46] and ROBPCA-AO [12].

For this experiment we only evaluate the semi-supervised approach of md-RPCA, sd-RPCA and kd-RPCA, considering that these semi-supervised algorithms presented the best results on the simulated experiment and taking into account the observed resilience of the semi-supervised approach when the training data is contaminated. Additionally, the semi-supervised approach only requires the robust subspace learning for training, what can indicate less computational cost for network anomaly detection on new observations.

The CTU-13 data set is very imbalanced, with contamination rate between 0.06% and 8.86%. Therefore we adopted an uniform contamination rate of 33% for this experiment, considering that the simulated experiment showed better results for contamination higher than 30% for m-RPCA and for the selected anomaly detection algorithms.

Table 5: Network anomaly detection from CTU-13 with 33% of contamination

Algorithm	10	11	12	15	15-2	15-3	16	16-2	16-3	17	18	18-2	19	Avg
md-RPCA	0.83	0.79	0.95	0.78	0.78	0.87	0.95	0.87	0.80	0.82	0.83	0.82	0.51	0.81
kd-RPCA	0.76	0.76	0.90	0.82	0.57	0.76	0.91	0.50	0.80	0.73	0.83	0.81	0.48	0.74
sd-RPCA	0.25	0.75	0.34	0.64	0.50	0.75	0.86	0.50	0.77	0.33	0.82	0.81	0.21	0.57
PCA	0.33	0.64	0.69	0.65	0.55	0.62	0.75	0.50	0.77	0.33	0.82	0.01	0.61	0.56
MCD	0.18	0.29	0.09	0.34	0.79	0.62	0.04	0.58	0.20	0.41	0.20	0.20	0.36	0.33
IF	0.36	0.34	0.09	0.21	0.40	0.44	0.16	0.34	0.34	0.41	0.12	0.16	0.46	0.29
LOF	0.15	0.14	0.13	0.17	0.29	0.22	0.29	0.38	0.25	0.24	0.00	0.04	0.38	0.21
KNN	0.05	0.17	0.01	0.03	0.33	0.23	0.01	0.25	0.03	0.12	0.00	0.00	0.24	0.11
ROBPCA-AO	0.01	0.07	0.00	0.05	0.38	0.11	0.05	0.32	0.03	0.09	0.07	0.09	0.21	0.11

The Table 3 present the F_1 of each algorithm for all scenarios of CTU-13, sorted by the average F_1 of each algorithm for all scenarios, which is presented in the last column.

It is possible to observe that md-RPCA, kd-RPCA and sd-RPCA overcome the remain algorithms in average results for all scenarios, according to the column **avg**. The sd-PRCA presented an average of 0.57 while md-RPCA obtained 0.81 and kd-RPCA 0.74 in average. The PCA based algorithm performed similar to sd-RPCA in average, with results of 0.56 and 0.57, respectively. However the results of PCA and sd-RPCA for scenarios 12, 18-2 and 19 presented a large variation.

The md-RPCA algorithm presented an anomaly detection rate higher than 0.78 for almost all scenarios, with exception to scenario 19, where the anomaly detection rate of md-RPCA was 0.51. The anomalies of the scenario 19 are peer-to-peer botnet traffic generated by nsys.ay, which are related to botnet synchronization and not to network attacks, what can explain the low network detection rate of all evaluated algorithms, where the largest F_1 was 0.61 achieved by PCA.

The md-RPCA showed the best anomaly detection for 10 scenarios of a total with 13 scenarios. For the scenario 15, the best result was obtained by kd-RPCA, for the 15-2 scenario the best result was for MCD, and PCA was the best algorithm for scenario 19. Even thought md-RPCA not be the best result for scenarios 15, 15-2 and 19, the results of md-RPCA are very close to the best results obtained for scenarios 15 and 15-2.

It is important to note that IF, KNN, LOF, MCD and ROBPCA-AO present

the worse results for network attack detection on all scenarios of the CTU-13 data set, with average of 0.29, 0.11, 0.21, 0.33 and 0.11 respectively. From these algorithms, only MCD presented high result for one scenario, which was 0.79 for 15-2.

The CTU-13 data set is very challenging for anomaly detection approaches, due to the high imbalance and large volume of flows. However, CTU-13 is one of the up-to-date data sets for network attack detection and is one that provides the data imbalance observed in real anomaly detection problems. Unfortunately, was not possible to observe Pareto or Lognormal distributions on features of CTU-13, even though the finds of researchers that highlight the fitting between these distributions and Internet traffic [8, 9].

The results of anomaly detection on CTU-13 reveals that m-RPCA algorithms obtain good results for botnet attack detection on contaminated data, overcoming widely adopted algorithms for outlier detection.

6. Conclusion and Future Works

This paper proposed the m-RPCA, which a framework based on distances of moments computed from a robust subspace learned by RPCA, for anomaly detection on imbalanced and skewed data. The m-RPCA can be divided into md-RPCA, sd-RPCA and kd-RPCA algorithms, to denote approaches of m-RPCA based on distances of mean, skewness and kurtosis, respectively.

We proposed approaches of m-RPCA for semi-supervised, contaminated semi-supervised and unsupervised methods of anomaly detection, and evaluated the anomaly detection score of m-RPCA for simulated data and for the CTU-13 data set, which is composed by network traffic of botnets, attacks and background flows.

The experimental results show that moment distances computed from robust subspace can improve the anomaly detection on skewed and imbalanced data set. The results also show that the m-RPCA can be adopted for network attack detection, with better results than widely adopted algorithms for outlier

detection on the CTU-13 data set.

Future research can be directed to evaluate the application of the proposed approaches to different data sets and to anomaly detection problems. The m-RPCA can also be extended to be online and to learn new subspace in an adaptive fashion, by means of new robust subspace learning from new observations or through robust subspace tracking. Future works can also consider to evaluate the sparse matrix $\boldsymbol{S}$ and its sparsity [51] for m-RPCA, to improve the accuracy and to be able to have element-wise anomaly detection.

References

[1] D. Networks, Bad bot report 2019: The bot arms race continues, accessed: 2019-06-15 (2019).
URL https://resources.distilnetworks.com/white-paper-reports/bad-bot-report-2019

[2] F. Catucci, M. Isbitski, R. Krikken, Protecting web applications and apis from exploits and abuse, accessed: 2019-06-15 (2019).
URL https://www.gartner.com/document/3907150

[3] P. Hevesi, Ddos: A comparison of defense approaches, accessed: 2019-06-15 (2019).
URL https://www.gartner.com/document/3907156

[4] J. Wang, I. C. Paschalidis, Botnet detection based on anomaly and community detection, IEEE Transactions on Control of Network Systems 4 (2) (2017) 392–404.

[5] A. Wang, W. Chang, S. Chen, A. Mohaisen, Delving into internet ddos attacks by botnets: Characterization and analysis, IEEE/ACM Trans. Netw. 26 (6) (2018) 2843–2855. doi:10.1109/TNET.2018.2874896.
URL https://doi.org/10.1109/TNET.2018.2874896

[6] N. Moustafa, J. Hu, J. Slay, A holistic review of network anomaly detection systems: A comprehensive survey, Journal of Network and Computer Applications 128 (2019) 33–55.

[7] A. Lakhina, M. Crovella, C. Diot, Mining anomalies using traffic feature distributions, in: ACM SIGCOMM Computer Communication Review, Vol. 35, ACM, 2005, pp. 217–228.

[8] T. Benson, A. Akella, D. A. Maltz, Network traffic characteristics of data centers in the wild, in: Proceedings of the 10th ACM SIGCOMM conference on Internet measurement, ACM, 2010, pp. 267–280.

[9] A. Leon-Garcia, Probability, statistics, and random processes for electrical engineering.

[10] C. Phua, D. Alahakoon, V. Lee, Minority report in fraud detection: Classification of skewed data, SIGKDD Explor. Newsl. 6 (1) (2004) 50–59. doi:10.1145/1007730.1007738.
URL http://doi.acm.org/10.1145/1007730.1007738

[11] H. He, E. A. Garcia, Learning from imbalanced data, IEEE Transactions on Knowledge & Data Engineering (9) (2008) 1263–1284.

[12] M. Hubert, P. Rousseeuw, T. Verdonck, Robust pca for skewed data and its outlier map, Computational Statistics & Data Analysis 53 (6) (2009) 2264–2274.

[13] C. Pascoal, M. R. De Oliveira, R. Valadas, P. Filzmoser, P. Salvador, A. Pacheco, Robust feature selection and robust pca for internet traffic anomaly detection, in: 2012 Proceedings IEEE INFOCOM, IEEE, 2012, pp. 1755–1763.

[14] C. Zhou, R. C. Paffenroth, Anomaly detection with robust deep autoencoders, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2017, pp. 665–674.

[15] S. Garcia, M. Grill, J. Stiborek, A. Zunino, An empirical comparison of botnet detection methods, computers & security 45 (2014) 100–123.

[16] O. Osanaiye, K.-K. R. Choo, M. Dlodlo, Distributed denial of service (ddos) resilience in cloud: review and conceptual cloud ddos mitigation framework, Journal of Network and Computer Applications 67 (2016) 147–165.

[17] M. Hubert, P. J. Rousseeuw, K. Vanden Branden, Robpca: a new approach to robust principal component analysis, Technometrics 47 (1) (2005) 64–79.

[18] M. H. Bhuyan, D. K. Bhattacharyya, J. K. Kalita, Network anomaly detection: methods, systems and tools, IEEE Communications Surveys & Tutorials 16 (1) (2014) 303–336.

[19] M. Ahmed, A. N. Mahmood, J. Hu, A survey of network anomaly detection techniques, Journal of Network and Computer Applications 60 (2016) 19–31.

[20] G. Gu, R. Perdisci, J. Zhang, W. Lee, Botminer: Clustering analysis of network traffic for protocol-and structure-independent botnet detection.

[21] D. Acarali, M. Rajarajan, N. Komninos, I. Herwono, Survey of approaches and features for the identification of http-based botnet traffic, Journal of Network and Computer Applications 76 (2016) 1–15.

[22] G. Gu, P. A. Porras, V. Yegneswaran, M. W. Fong, W. Lee, Bothunter: Detecting malware infection through ids-driven dialog correlation., in: USENIX Security Symposium, Vol. 7, 2007, pp. 1–16.

[23] S. Khattak, Z. Ahmed, A. A. Syed, S. A. Khayam, Botflex: A community-driven tool for botnet detection, Journal of Network and Computer Applications 58 (2015) 144–154.

[24] M. Tavallaee, E. Bagheri, W. Lu, A.-A. Ghorbani, A detailed analysis of the kdd cup 99 data set, in: Proceedings of the Second IEEE Symposium

on Computational Intelligence for Security and Defence Applications 2009, 2009.

[25] H. Ringberg, A. Soule, J. Rexford, C. Diot, Sensitivity of pca for traffic anomaly detection, in: ACM SIGMETRICS Performance Evaluation Review, Vol. 35, ACM, 2007, pp. 109–120.

[26] C. Callegari, L. Gazzarrini, S. Giordano, M. Pagano, T. Pepe, A novel pca-based network anomaly detection, in: 2011 IEEE International Conference on Communications (ICC), IEEE, 2011, pp. 1–5.

[27] Y.-J. Lee, Y.-R. Yeh, Y.-C. F. Wang, Anomaly detection via online oversampling principal component analysis, Knowledge and Data Engineering, IEEE Transactions on 25 (7) (2013) 1460–1470. doi:10.1109/TKDE.2012.99.

[28] T. P. Vieira, D. F. Tenório, J. P. C. da Costa, E. P. de Freitas, G. Del Galdo, R. T. de Sousa Júnior, Model order selection and eigen similarity based framework for detection and identification of network attacks, Journal of Network and Computer Applications 90 (2017) 26–41.

[29] E. J. Candès, X. Li, Y. Ma, J. Wright, Robust principal component analysis?, Journal of the ACM (JACM) 58 (3) (2011) 11.

[30] N. Vaswani, T. Bouwmans, S. Javed, P. Narayanamurthy, Robust subspace learning: Robust pca, robust subspace tracking, and robust subspace recovery, IEEE signal processing magazine 35 (4) (2018) 32–55.

[31] G. Lerman, T. Maunu, An overview of robust subspace recovery, Proceedings of the IEEE 106 (8) (2018) 1380–1410.

[32] Y. Cherapanamjeri, P. Jain, P. Netrapalli, Thresholding based efficient outlier robust pca, arXiv preprint arXiv:1702.05571.

[33] Rad: Time series anomaly detection, https://github.com/Netflix/Surus/, accessed: 2019-06-15 (2018).

[34] P. C. Mahalanobis, On the generalized distance in statistics, Proceedings of the National Institute of Sciences (Calcutta) 2 (1936) 49–55.

[35] S. Garcıa, Identifying, modeling and detecting botnet behaviors in the network, Unpublished doctoral dissertation, Universidad Nacional del Centro de la Provincia de Buenos Aires.

[36] J. Wright, A. Ganesh, S. Rao, Y. Peng, Y. Ma, Robust principal component analysis: Exact recovery of corrupted low-rank matrices via convex optimization, in: Advances in neural information processing systems, 2009, pp. 2080–2088.

[37] Z. Lin, M. Chen, Y. Ma, The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices, arXiv preprint arXiv:1009.5055.

[38] X. Yuan, J. Yang, Sparse and low-rank matrix decomposition via alternating direction methods, preprint 12 (2009) 2.

[39] P. M. G. Reis, J. P. C. da Costa, R. K. Miranda, G. Del Galdo, Audio authentication using the kurtosis of esprit based enf estimates, in: 2016 10th International Conference on Signal Processing and Communication Systems (ICSPCS), IEEE, 2016, pp. 1–6.

[40] D. Zwillinger, S. Kokoska, CRC standard probability and statistics tables and formulae, Crc Press, 1999.

[41] P. J. Rousseeuw, Least median of squares regression, Journal of the American statistical association 79 (388) (1984) 871–880.

[42] P. J. Rousseeuw, K. V. Driessen, A fast algorithm for the minimum covariance determinant estimator, Technometrics 41 (3) (1999) 212–223.

[43] Y. Zhao, Z. Nasrullah, Z. Li, Pyod: A python toolbox for scalable outlier detection, arXiv preprint arXiv:1901.01588.

[44] S. Arlot, A. Celisse, et al., A survey of cross-validation procedures for model selection, Statistics surveys 4 (2010) 40–79.

[45] D. M. Powers, Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation.

[46] M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn, L. Chang, A novel anomaly detection scheme based on principal component classifier, Tech. rep., MIAMI UNIV CORAL GABLES FL DEPT OF ELECTRICAL AND COMPUTER ENGINEERING (2003).

[47] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, R. C. Williamson, Estimating the support of a high-dimensional distribution, Neural computation 13 (7) (2001) 1443–1471.

[48] M. M. Breunig, H.-P. Kriegel, R. T. Ng, J. Sander, Lof: identifying density-based local outliers, in: ACM sigmod record, Vol. 29, ACM, 2000, pp. 93–104.

[49] F. Angiulli, C. Pizzuti, Fast outlier detection in high dimensional spaces, in: European Conference on Principles of Data Mining and Knowledge Discovery, Springer, 2002, pp. 15–27.

[50] F. T. Liu, K. M. Ting, Z.-H. Zhou, Isolation forest, in: 2008 Eighth IEEE International Conference on Data Mining, IEEE, 2008, pp. 413–422.

[51] K. Liu, F. Roemer, J. P. C. da Costa, J. Xiong, Y.-S. Yan, W.-Q. Wang, G. Del Galdo, Tensor-based sparsity order estimation for big data applications, in: 2017 25th European Signal Processing Conference (EUSIPCO), IEEE, 2017, pp. 648–652.