ASTES

Advances in Science, Technology and Engineering Systems Journal
Vol. 5, No. 3, 249-258 (2020)
www.astesj.com
Special Issue on Multidisciplinary Innovation in Engineering Science & Technology

ASTES Journal
ISSN: 2415-6698

Improved Nonlinear Fuzzy Robust PCA for Anomaly-based Intrusion Detection

Amal Hadri[*,1], Khalid Chougdali[2], Raja Touahni[1]

[1]*LASTID Laboratory, Faculty of Science, Ibn tofail University, Kenitra, Morocco*

[2]*GREST Research Group, National School of Applied Sciences (ENSA), Kenitra, Morocco*

ARTICLE INFO

Article history:
Received: 05 February, 2020
Accepted: 10 May, 2020
Online: 28 May, 2020

Keywords:
KDDCup99
Principal Component Analysis
PCA
NFRPCA
IDS
Feature extraction methods
NSL-KDD

ABSTRACT

Among the most popular tools in security field is the anomaly based Intrusion Detection System (IDS), it detects intrusions by learning to classify the normal activities of the network. Thus if any abnormal activity or behaviour is recognized it raises an alarm to inform the users of a given network. Nevertheless, IDS is generally susceptible to high false positive rate and low detection rate as a result of the huge useless information contained in the network traffic employed to build the IDS. To deal with this issue, many researchers tried to use a feature extraction methods as a pre-processing phase. Principal Component Analysis (PCA) is the excessively popular method used in detection intrusions area. Nonetheless, classical PCA is prone to outliers, very sensitive to noise and also restricted to linear principal components. In the current paper, to overcome that we propose a new variants of the Nonlinear Fuzzy Robust PCA (NFRPCA) utilizing the popular data sets KDDcup99 and NSL-KDD. The results of the conducted experiments demonstrated that the proposed approaches is more effective and gives a promising efficiency in comparison to NFRPCA and PCA.

1 Background

Thanks to the major shift in technology tools in the twenty first century, the complexity of network security has greatly increased, which gave birth to highly developed attacks. There are several traditional security methods like firewalls, data encryption and user authentication. Those techniques are insufficient to protect the network systems against all the existing threats. As a consequence, they may be less effective in detecting several dangerous attacks. Therefore, we need to strengthen our systems by adding more powerful systems such as intrusion detection system (IDS). The IDS protects the network systems by preventing the eventual damages that could be caused by an intrusion. Commonly, there is 2 principal categories of IDS, misuse-based and anomaly-based. The misuse-based method aim to classify an attack via comparing its signature with the attacks currently existing in a database of signatures of attacks and produce an alarm if any malicious activity is detected. The most two well-known misuse detection methods are STAT [1] and Snort [2]. This technique has proved its effectiveness in detecting the attacks stored in the datasets but it can not detects new intrusions or attacks and maintaining the databases is very expensive. Hence anomaly-based detection was initiated by Anderson [3] & Denning [4], the fundamental idea behind this concept is to specify the normal behaviour or model and generate an alarm if the difference between an observation and the defined model surpasses a threshold already defined. The uniqueness of this concept is its capability to categorize new and unusual intrusions.

Nonetheless, the current network traffic data, which are usually tremendous, are in fact a big challenge to anomaly based IDS. This type of traffic may decrease the whole detection mechanism and lead frequently to a falsified classification accuracies. This kind of huge dataset often have redundant and noisy data, that can be very difficult to model.

To address that, many feature extraction techniques have been used to increase network IDS efficiency. For instance, the paper [5] used a Discrete Differential Evolution to recognize the most important features. The detection accuracies were enhanced significantly. Likewise, the work [6] presented an IDS that can detect several attacks by exploring just a small number of features, the algorithm utilized is called Ant Colony Optimization algorithm. In [7] the authors used a feature selection method called cuttlefish

[*]Corresponding Author Amal Hadri, LASTID Laboratory, Faculty of Science, Ibn tofail University, Kenitra, Morocco, amal.hadri@uit.ac.ma

which suppress the noisy and redundant data and simultaneously guarantee the quality of data. The authors of [8] proposed to utilize the Principal Component Analysis (PCA) and Fuzzy Principal Component Analysis (FPCA) as a pre-processing step, before applying the k nearest neighbor (KNN) classifier, the same authors suggest in another publication [9] an improved variant method called Robust Fuzzy PCA. The acquired results show the promising performance of the technique proposed with regard to network attacks detection, as well as false alarms reduction.

Neverthless, PCA [10]-[12] and its linear variants are known to be sensitive to noise and outliers, which can impact on the deriving principal component (PCs) [13], therefore they effect as well the results of classification. In addition to that, PCA allows uniquely a linear dimensionality reduction [14]. Hence, in the case of complicated structures like nonlinear structures, the data will not be correctly expressed in a linear space, linear variants of PCA will not be the optimal solution. To deal with this issue, NFRPCA (Nonlinear Fuzzy Robust PCA) [15] was suggested to calculate PCs by utilizing a non-linear technique.

Nevertheless, this method is based on the L_2-norm that is highly sensitive to outliers and it also squares the error, and so the model can have a much bigger error. So as to tackle this issue, we introduce a new variant of NFRPCA called L_p-norm NFRPCA. Note that this paper is an extension of work originally published in 2018 IEEE 5th International Congress on Information Science and Technology (CiSt) [16], in that work we suggested a variant of NFRPCA employing L_1-norm rather than the classical Euclidian norm. In this paper, we propose another variant of NFRPCA using L_p-norm, we conducted new experiments besides the ones previously proposed in [16].

The remainder of this paper is structured as follows: Section 2 deliver a brief presentation of PCA, Section 3 will present an overview of NFRPCA. We present the suggested techniques in Section 4, Section 5 is dedicated to give shortly an overview of the two popular datasets namely KDDcup99 and NSL-KDD. Section 6 presents the conducted experiments and discuss the results and conclusions are summarized in section 7.

2 Principal Component Analysis Method

Principal Component Analysis (PCA) [17] is as an exploratory data analysis tool that involves a dataset with observations on variables, it was employed extensively in several research areas. The principal concept of PCA is to change data into a downsized form and preserve most of the initial variance from the original data at the same time. In other terms, PCA major role is to change variables n that were correlated into uncorrelated state d, the uncorrelated variables d are usually called the principal components (PCs) [14] [18].

Consider we already have a data set of M vectors $v_1,\ v_2,\ v_3, ...,\ v_M$ where each vector is represented by N features. To obtain the PCs we comply to the strategy explained through the steps underneath:

- Compute the mean μ of the data set

$$\mu = \frac{1}{M}\sum_{i=1}^{M} v_i \tag{1}$$

- Determine the deviation from the mean

$$\theta_i = v_i - \mu \tag{2}$$

- Compute the covariance matrix of the corresponding data set:

$$C_{n*n} = \frac{1}{M}\sum_{i=1}^{M} \theta_i {\theta_i}^T = \frac{1}{M} A A^T \tag{3}$$

where $A = [\theta_1, \theta_2, \theta_3, ..., \theta_n]$

- Assume that U_k is the k^{th} eigenvector of C, λ_k the related eigenvalue and let's say that the $U_{n*d} = [U_1, U_2, ..., U_d]$ is the matrix of these eigenvectors. Hence:

$$CU_k = \lambda_k U_k \tag{4}$$

- Sort the Eigenvalues in decreasing order, hence pick the eigenvectors (known as principal components PC_i)that have the largest Eigenvalues. We can compute the number of PCs that we could keep as follow:

$$\tau = \frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \tag{5}$$

- Consider t as a new sample column vector, t is projected on the reduced subspace covered by the PC_i :

$$y_i = U_i^T t \tag{6}$$

3 Nonlinear Fuzzy Robust pca (nfrpca) Method

The Nonlinear fuzzy robust principal component analysis employed in the current paper was suggested initially by Luukka in [15]. It was inspired from the robust principal component techniques that Yang & Wang proposed in [19] that fundamentally introduced by Xu & Yuilles techniques [20] where PCA learning rules are associated to energy functions and they proposed a cost function in regard to outliers. In Yang & Wang's proposed methods the cost function was modified to be fuzzy and it involved Xu & Yuilles techniques as a particular case. We introduce briefly these methods in this section. Xu and Yuille [20] suggested an objective function, based on $u_i \in \{0, 1\}$:

$$E(U, w) = \sum_{i=1}^{n} u_i e(x_i) + \eta \sum_{i=1}^{n} 1 - u_i \tag{7}$$

The variables are defined as follows: η is the threshold, $U = \{u_i | i = 1, , n\}$ is the membership set, & $X = \{x_1, x_2, ..., x_n\}$ is the dataset. The principal objective is to minimize $E(U, w)$ with regard of u_i & w. It should be noted that w is a continuous variable and u_i is a binary

variable that engender optimization challenging to resolve with a gradient descent technique. To solve the problem employing the gradient descent technique the minimization problem is simplified into maximization of Gibbs distribution as underneath:

$$P(U,w) = \frac{exp(-vE(U,w))}{Z} \tag{8}$$

Where Z is the partition function confirming $\sum_U \int_w P(U,w) = 1$. The measure $e(x_i)$ might be, e.g. one of the below functions

$$e_1(x_i) = \left\| x_i - w^T x_i w \right\|^2 \tag{9}$$

$$e_2(x_i) = \|x_i\|^2 - \frac{\left\|w^T x_i\right\|^2}{\|w\|^2} = x_i^T x_i - \frac{w^T x_i x_i^T w}{w^T w} \tag{10}$$

To minimize $E_1 = \sum_{i=1}^n e_1(x_i)$ and $E_2 = \sum_{i=1}^n e_2(x_i)$, the gradient descent rules are

$$w^{new} = w^{old} + \alpha_t[y(x_i - u) + (y - v)x_i] \tag{11}$$

$$w^{new} = w^{old} + \alpha_t(x_i y - \frac{w}{w^T w} y^2) \tag{12}$$

Where $y = w^T x_i$, $u = yw$,$v = w^T u$ and α_t is the learning rate. The nonlinear case of PCA was presented as underneath :

$$e_3(x_i) = \left\| x_i - w^T g(y) \right\| \tag{13}$$

And $y = x_i\, w$ and g could be selected as nonlinear functions. The weight updating in this situation is

$$w^{new} = w^{old} + \alpha_t(x_i e^T w^{old} F + e_3(x_i) g(y)) \tag{14}$$

And: $F = \frac{d}{dy}(g(y))$.

The cost function proposed by Yang and Wang:

$$E = \sum_{i=1}^{n} u_i^{m_1} e(x_i) + \eta \sum_{i=1}^{n} (1 - u_i)^{m_1} \tag{15}$$

subject to $u_i \in [0,1]$ and $m_1 \in [1, \infty)$. Now u_i is the membership value of x_i associated to the data cluster and $(1 - u_i)$ is the membership value of x_i associated to the noise cluster and m_1 is the fuzziness variable. And $e(x_i)$ is the error between the class center and x_i.

As u_i is a continuous variable now, the complexity of an amalgamation of continuous and discrete optimization could be obviated and the gradient descent technique can be employed.

The gradient of E(15) is calculated with regard to u_i.

By choosing $\frac{\delta E}{\delta u_i} = 0$, we obtain

$$u_i = \frac{1}{1 + \left(\frac{e(x_i)}{\eta}\right)^{\frac{1}{(m_1 - 1)}}} \tag{16}$$

Replacing this membership back, we get

$$E = \sum_{i=1}^{n} \left(\frac{1}{1 + \left(\frac{e(x_i)}{\eta}\right)^{\frac{1}{m_1 - 1}}} \right)^{m_1 - 1} e(x_i) \tag{17}$$

The gradient with regard to w would be

$$\frac{\partial E}{\partial W} = \left(\frac{1}{1 + \left(\frac{e(x_i)}{\eta}\right)^{\frac{1}{(m_1 - 1)}}} \right)^{m_1} \left(\frac{\partial e(x_i)}{\partial w} \right) \tag{18}$$

Consider

$$\beta(x_i) = \left(\frac{1}{1 + \left(\frac{e(x_i)}{\eta}\right)^{\frac{1}{(m_1 - 1)}}} \right)^{m_1} \tag{19}$$

where m_1 is the fuzziness variable. If $m_1 = 1$, the fuzzy membership downsizes into hard membership and could be picked following the concept:

$$u_i = \begin{cases} 1 \; if \;\; e(x_i) < \eta \\ 0 \;\; otherwise \end{cases}$$

right now η is a though threshold in this situation. The setting of m_1 has no rule. We sum up NFRPCA steps in Algorithm1.

Algorithm 1 NFRPCA algorithm

Step 1: At first set the count of iteration $t = 1$, bound of iteration T, learning coefficient $\alpha_0 \in (0, 1]$ soft threshold η to a very small positive value then initialize in random way the weight w.

Step 2: As long as t is smaller than T, do steps 3-9.

Step 3: Calculate $\alpha_t = \alpha_0(1 - t/T)$, set $i = 1$ and $\sigma = 0$.

Step 4: As long as i is smaller than n, do steps 5-8

Step 5: Calculate $y = w^T x_i$, $u = yw$ and $v = w^T u$.

Step 6: Calculate $g(y)$, $F = \frac{d}{dy}(g(y))$, $e_3(x_i) = x_i - w^{old} g(y)$, then the weight is updated:

$w^{new} = w^{old} + \alpha_t(x_i e^T w^{old} F + e_3(x_i) g(y))$

In [15] $g(y)$ was selected to be a sigmoid function such as $g(y) = tanh(10y)$, F is the first derivative of $g(y)$.

4 The Proposed Methods

4.1 Nonlinear Fuzzy Robust L_1-norm PCA method

We can clearly remark that, NFRPCA technique utilize an Euclidian norm for computing the reconstruction error in (13), and it is widely recognized that the Euclidian norm usually squares the reconstruction error, thus the approach have a much bigger error. Consequently, this could falsify the results and deteriorate the quality of solutions. For the sake of addressing this issue, the paper [16] suggest utilizing L_1-norm to compute the reconstruction error.

The reconstruction error equation could be re-written as below:

$$e_3'(x_i) = \left\| x_i - w'^T g(y') \right\|_1 \tag{20}$$

$y' = x_i w'$ and g could be picked as nonlinear functions. Here the weight updating is

$$w^{new'} = w^{old'} + \alpha_t(x_i e^{T'} w^{old'} F' + e_3'(x_i) g(y')) \tag{21}$$

Where :$F' = \frac{d}{dy'}(g(y'))$.
Equally as in algorithm 1, we utilize updating weight to compute the PCs.

Algorithm 2 L_1-NFRPCA algorithm

Step 1: At first set the count of iteration $t = 1$, bound of iteration T, learning coefficient $\alpha_0 \in (0, 1]$ soft threshold η to a very small positive value then initialize in random way the weigh w.
Step 2: As long as t is smaller than T, do steps 3-9.
Step 3: Calculate $\alpha_t = \alpha_0(1 - t/T)$, set $i = 1$ and $\sigma = 0$.

Step 4: As long as i is smaller than n, do steps 5-8

Step 5: Calculate $y' = w'^T x_i$, $u' = y'w'$ and $v' = w'^T u'$.

Step 6: Calculate $g(y')$, $F' = \frac{d}{dy'}(g(y'))$, $e'_3(x_i) = x_i - w'^{old} g(y')$, then the weight is updated: $w'^{new} = w'^{old} + \alpha_t(x_i e'^T w'^{old} F' + e'_3(x_i) g(y'))$

In the proposed algorithm $g(y')$ was picked to be sigmoid like the function $g(y') = tanh(10y')$, & F' is the first derivative of $g(y')$. We use the term L_1-norm NFRPCA to refer to this technique in the rest of this paper.

4.2 Nonlinear Fuzzy Robust L_p-norm PCA method

The technique that we suggest is L_p-norm NFRPCA where we propose to substitute the L_2-norm with generalized L_p-norm in the computation of the reconstruction error, in order to minimize the large error that L_2-norm can cause. The reconstruction error equation would be re-written as follow:

$$e^{"}_3(x_i) = \left\| x_i - w^{"T} g(y") \right\|_p \tag{22}$$

Where $0 > p >= 2$.
And $y" = x_i w"$ and g could be selected as nonlinear function. Here the weight updating is

$$w^{new"} = w^{old"} + \alpha_t(x_i e^{T"} w^{old"} F" + e^{"}_3(x_i) g(y")) \tag{23}$$

Where: $F" = \frac{d}{dy"}(g(y"))$.

In the same way as in algorithm 1, we use the updating weight to compute the PCs. Finally the main steps of the proposed method is summarized in Algorithm 3.

Algorithm 3 L_p-norm NFRPCA algorithm

Step 1: At first set the count of iteration $t = 1$, bound of iteration T, learning coefficient $\alpha_0 \in (0, 1]$ soft threshold η to a very small positive value then initialize in random way the weight $w"$.
Step 2: As long as t is smaller than T, do steps 3-9.
Step 3: Calculate $\alpha_t = \alpha_0(1 - t/T)$, set $i = 1$ and $\sigma = 0$.

Step 4: As long as i is smaller than n, do steps 5-8

Step 5: Calculate $y" = w^{"T} x_i$, $u" = y"w"$ and $v" = w^{"T} u"$.

Step 6: Calculate $g(y")$, $F" = \frac{d}{dy"}(g(y"))$, $e"_3(x_i) = x_i - w"^{old} g(y")$, then the weight is updated: $w"^{new} = w"^{old} + \alpha_t(x_i e"^T w"^{old} F" + e^{"}_3(x_i) g(y"))$

The function $g(y")$ was picked to be sigmoid function, such as $g(y") = tanh(10y")$, and $F"$ is the first derivative of $g(y")$. We use the term L_p-norm NFRPCA to refer to this technique in the rest of this paper.

5 The Simulated Datasets and its Preprocessings

5.1 Datasets

In the experiments we conducted, the popular public intrusion data sets were used, they are called KDDcup99 and NSL-KDD. We present them shortly in the next subsections.

5.1.1 KDDCup99 Dataset

KDDCup99 [21, 22] is a dataset that contains many TCPdump raws, that was captured during 9 weeks. This dataset was prepared and still conducted by Intrusion Detection Evaluation Program called DARPA. The main aim of KDDCup99 is to establish a generalized data set to examine researchs in intrusion detection field.

The training data set represents 4 gigabytes of data compressed (most of it is binary TCP dump), it contains 4,898,431 records, while the test dataset contains around 311,029 connection records and each record in dataset contains exactly 41 features. The attacks existing in KDDCup99 are categorized as underneath:

- Denial of Service (DoS): cyber-attack where the perpetrator attempt to consume network or machine resources to make it unavailable or limited to its legitimate users.
- Remote to Local (R2L): Where the attacker control the remote machine pretending that he is a user of the system, by exploiting the system vulnerabilities.
- User to Root (U2R): by exploiting the vulnerabilities and the flaws existing in a machine or on a network, the hacker tries to start accessing from a normal user account as to get the root access privilege to the system.
- Probing: The intruder tries to collect useful information of all services and machines existing in the same network to exploit it later.

5.1.2 NSL-KDD Dataset

The NSL-KDD [23] data set has been created to alleviate a couple of the major shortcoming of the KDDCup99 data set. This new version has some improvement compared to KDDCup99 data set and has solved a few of its fundamental issues. The main advantages of NSL-KDD are as follow:

- It suppressed redundant instances in the trainining set.
- The test dat set does not contain replicate records.

- The current version allow us to use the whole data set. Correspondingly, the random selection method will be needless because of the reduction of the number of instances in both train set and test set. Consequently, the accuracy and consistency of the evaluation and review of research works will increase.

- Every complexity level group involve a couple of instances that is oppositely corresponding to the percentage of instances in the KDDCup99 dataset. Therefore, we obtain more realistic examination of various machine learning techniques.

5.2 Normalization Phase

The normalization phase is more than important, because it allow us to apply the techniques cited above on the data set in a correct manner. To perform effectively this phase, we replaced the discrete values with continuous values for all the discrete attributes existing in the data sets through the same process used in [10], the process is briefly clarified as follow: for every attribute y that acceptes x variant values. The attribute y is illustrated by x coordinates contains zeros and ones. E.g., the attribute of the protocol type, that acceptes three, values i.e. tcp, udp or icmp. Utilizing this logic, these values are modified to the following coordinates (1,0,0), (0,1,0) or (0,0,1).

6 Experiments and Discussion

In the current part of this paper, several experiments were performed to examine the efficiency of the suggested method utilizing KDDcup99 and NSL-KDD databases, for the sake of evaluating the effectiveness of the suggested technique. We compute the measures: detection rate (DR), F-measure and false positive rate (FPR) described underneath:

$$DR = \frac{TP}{TP + FN} * 100 \tag{24}$$

$$FPR = \frac{FP}{FP + TN} * 100 \tag{25}$$

$$FMeasure = \frac{2 * TP}{2 * TP + FP + FN} * 100 \tag{26}$$

When :

- An intrusion is successfully predicted we call it a True positives (TP).

- An intrusion is wrongly predicted we call it a False negatives (FN).

- A normal connection is wrongly predicted we call it a False positive (FP).

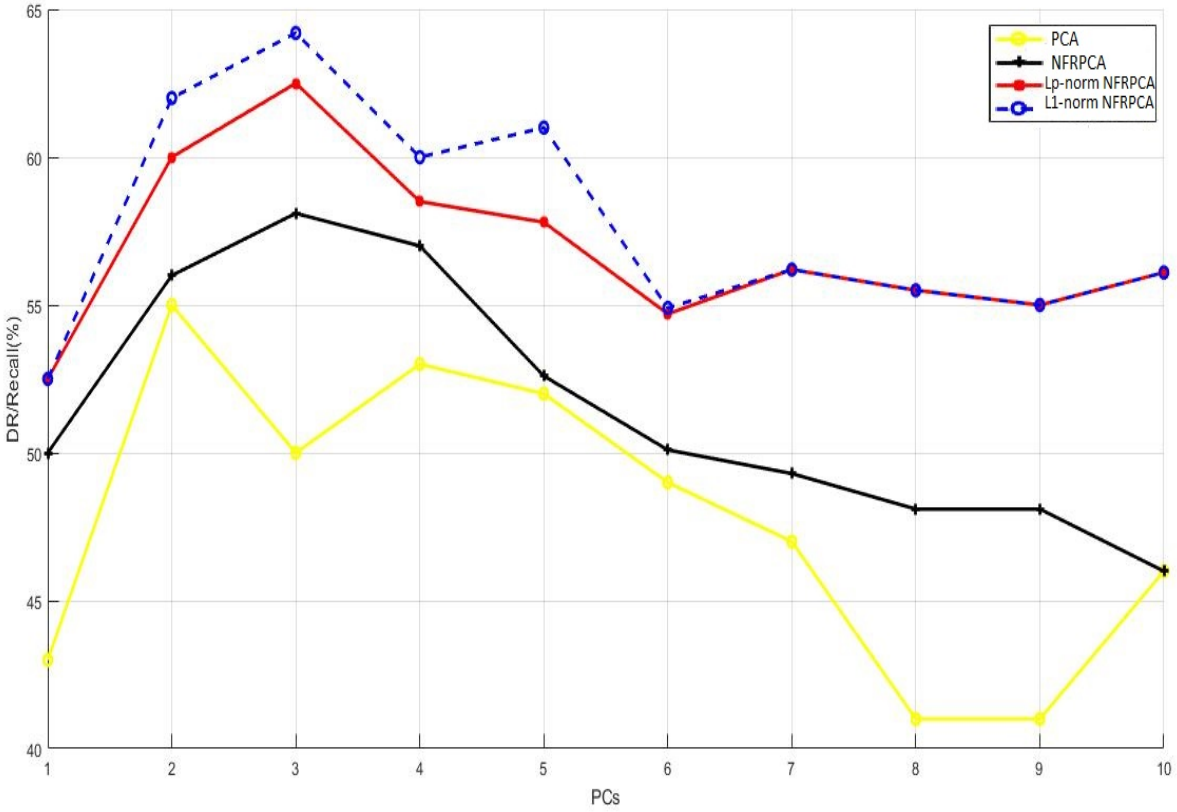

Figure 1: detection rate vs. PCs using KDDcup99 dataset

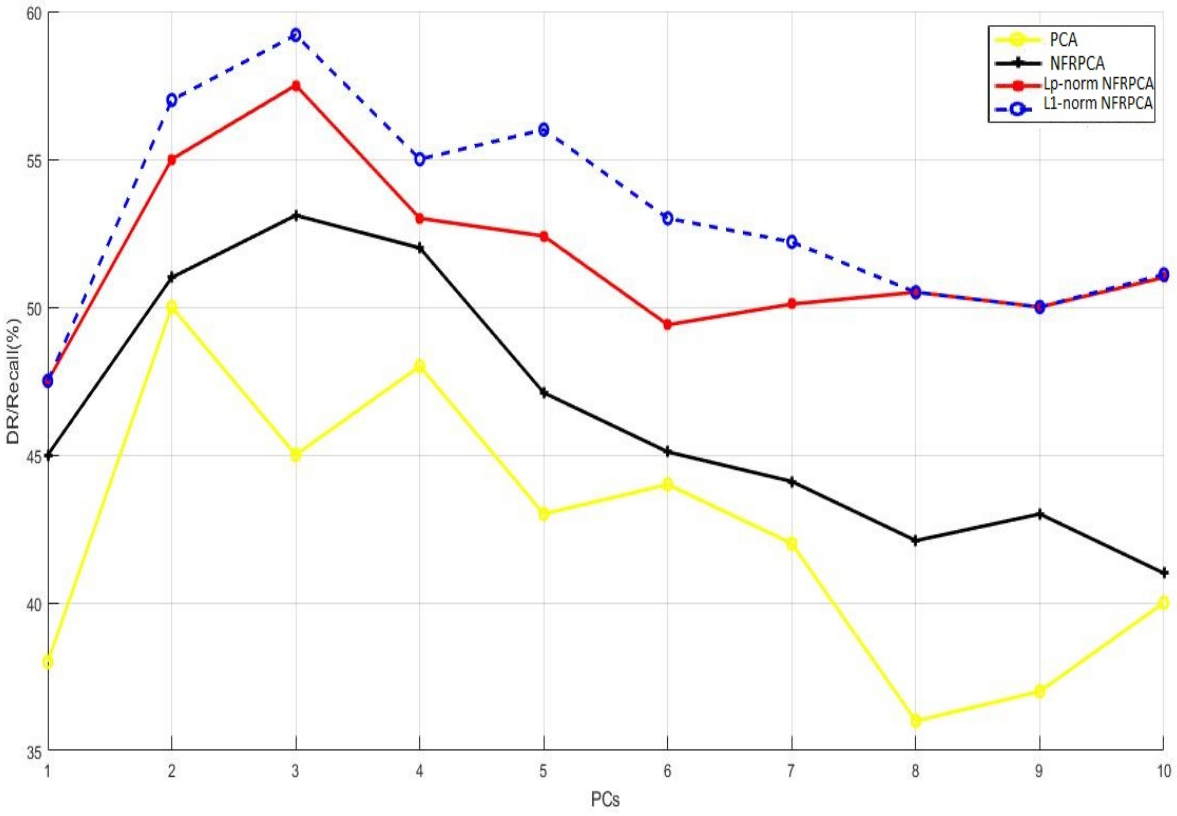

Figure 2: detection rate vs. PCs using NSL-KDD dataset

- A normal connection is correctly predicted we call it a True negatives (TN).

The classifier that was utilized in all our conducted experiments is the nearest neighbor, and for the sake of obtaining more trustworthy results we computed the average of twenty runs. The highest robust feature extraction technique must have the highest DR and F-measure rates and as much as possible the lowest FPR rate.
The simulation settings used in our first experiments are as follow: for the training sample we randomly selected 1000 normal, 100 DOS, 50 U2R, 100 R2L and 100 PROBE as for the test sample we have this structure: 100 normal data, 100 DOS data, 50 U2R data, 100 R2L data and 100 PROBE also choosed randomly using the test database. So as to both KDDcup99 and NSL-KDD data sets the settings of the simulation are similar. Also the value of p is set to 0.5 in all our experiments.

During our first experiments and to choose the ideal number of prinicipal components(PCs) for all the feature extraction techniques which helps drastically to raise F-measure and detection rate (DR) and decrease FPR, examine and make a comparaison between PCA, NFRPCA, L_1-NFRPCA and L_p-norm NFRPCA. To achieve that, we have performed PCA, NFRPCA ,L_1-NFRPCA and L_p-norm NFRPCA to train our model first. Consequently, we obtained the PCs. The number of principal components PCs represent the dimension of the new downsized samples. Furthermore, the test sample is proojected on the new downsized subspace established via the PCs.

The aim of our first experiment is to compute the measures detection rate, F-measure and false positive rate for each single principal component through choosing 10 of 41 principal component and changing their number iteratively throughout the test. We can observe clearly in the Figure.1 and Figure.2 in both datasets KDDCup99 and NSLKDD the L_1-NFRPCA and L_p-norm NFRPCA takes the lead over the original NFPRCA and the linear PCA which

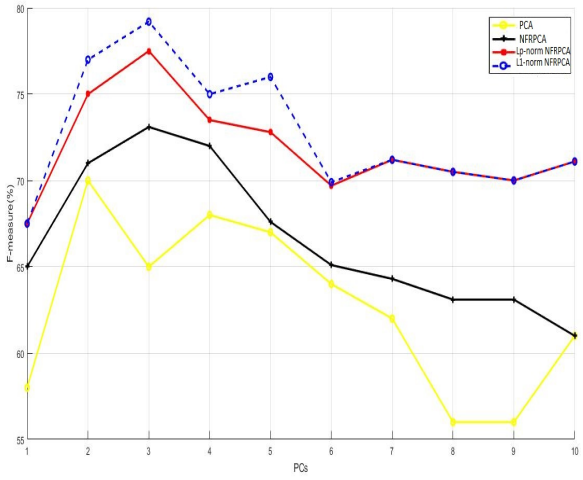

Figure 3: F-measure vs. PCs using KDDcup99 dataset

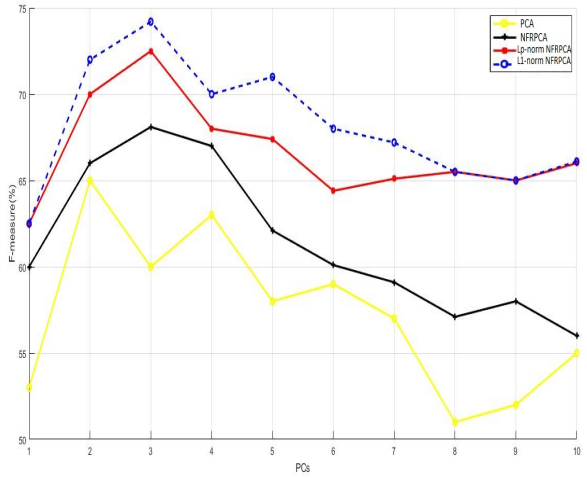

Figure 4: F-measure vs. PCs using NSL-KDD dataset

Following the previous concept, we computed the F-measure by increasing the number of PCs. As we can see from the Figure.3 and Figure.4, for both datasets the L_1-NFRPCA and L_p-norm NFRPCA got the highest values for the F-measure comparing to the original NFRPCA and the classical PCA which support our previous results.

And so as to calculate the false positive rate (FPR), all the algorithms cited above were implemented, but only the proposed algorithms (L_1-NFRPCA and L_p-norm NFRPCA) who has the lowest FPR for both datasets, as we see clearly in the figures (Figure 5 and Figure 6).

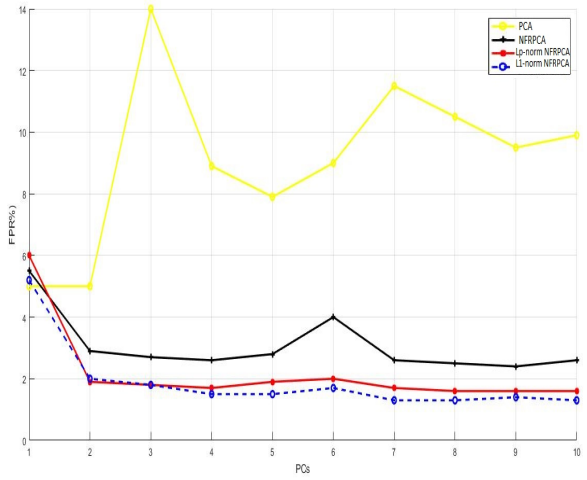

Figure 5: FPR vs. PCs using KDDcup99 dataset

Table 1: DR of Attacks for PCA, NRFPCA, L_1-NRFPCA and L_p-norm NFRPCA using KDDCup99 dataset

	Method	DOS	U2R	R2L	Probing
DR (%)	PCA	68,6656	8,6923	4,7734	92,1121
	NRFPCA	72,1123	14,4615	4,1165	90,2345
	L_1-NRFPCA	73,1993	15,9815	4,1775	91,8325
	L_p-norm NRFPCA	74,2314	16,1111	4,5556	92,1211

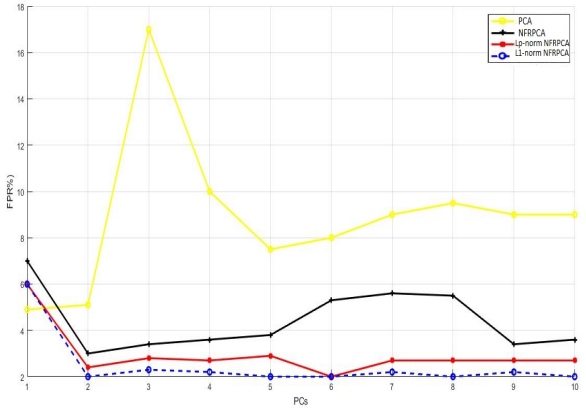

Figure 6: FPR vs. PCs using NSL-KDD dataset

In the second stage of our experiments, we intend to examine all the techniques cited above under a wide range of different training dimensionnality, and examine their impact on the DR, FPR, and F-measure. To achieve that, the structure of the test data set was kept intact by fixing it at 100 normal connections, 100 DOS, 50 U2R, 100 R2L, and 100 PROBE.

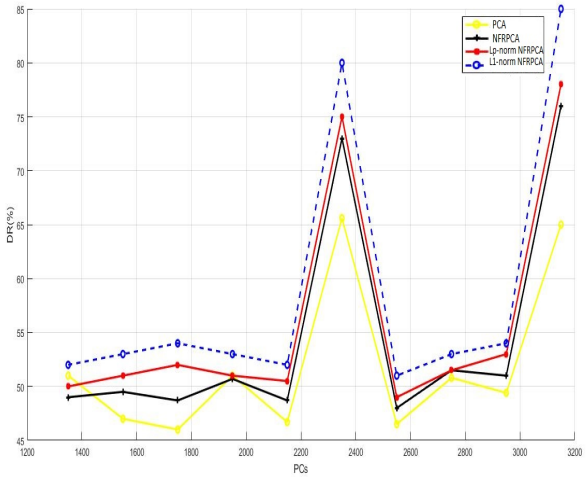

Figure 7: detection rate vs. Training data using KDDcup99 dataset

Concerning DR, Figure.7 and Figure.8 assert that the proposed methods produce a detection rate higher than the original ones. It proves that the methods are very powerful in differentiating between normal connections and attacks.

In Figure.9, Figure.10, we can clearly see that the L_1-NFRPCA achieve at least 5% improvement over L_p-norm NFRPCA, 10% over original NFRPCA and the classical PCA, the new approaches surpasses permanently the original techniques. In terms of FPR,the Figure.11 and Figure.12 show that the L_1-NFRPCA still gives the lowest FPR even under different dimensionnality. These results support the great capability of the new approaches to classify the connections autonomously of the training samples size.

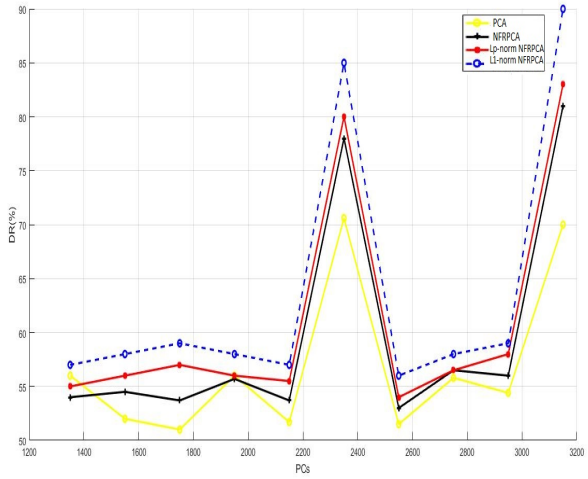

Figure 8: detection rate vs. Training data using NSL-KDD dataset

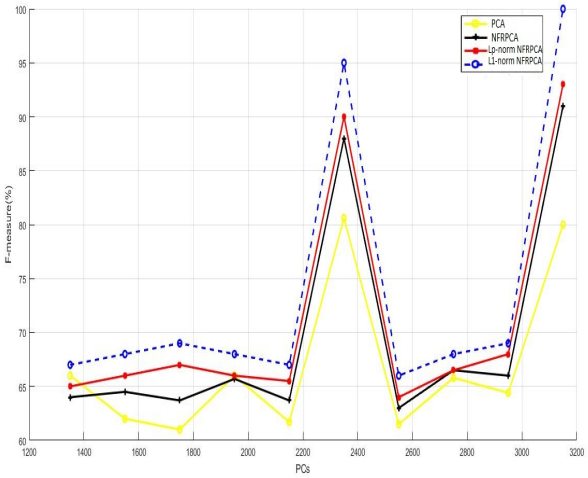

Figure 9: F-measure vs. Training data using KDDcup99 dataset

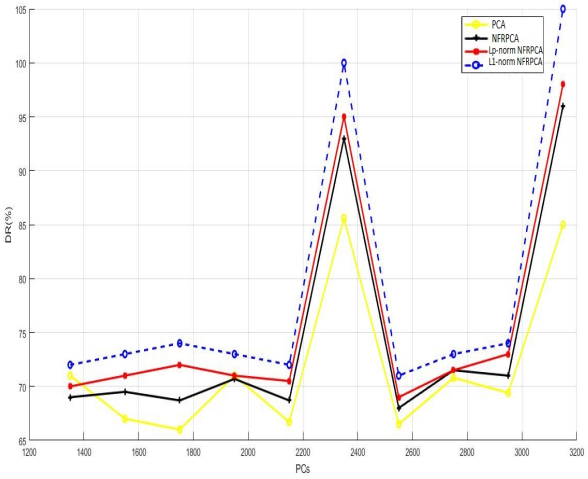

Figure 10: F-measure vs. Training data using NSL-KDD dataset

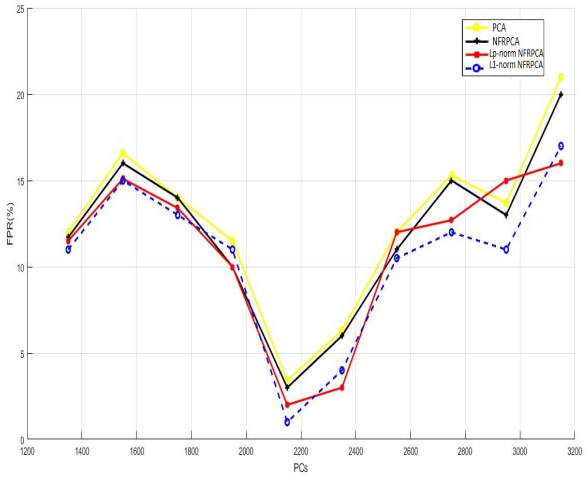

Figure 11: FPR vs. Training data using KDDcup99 dataset

Table 2: DR of Attacks for PCA, NRFPCA, L_1-NRFPCA and L_p-norm NFRPCA using NSL-KDD dataset

	Method	DOS	U2R	R2L	Probing
DR (%)	PCA	67,5546	7,9723	4,8872	93,1121
	NRFPCA	71,1211	13,6415	4,1435	90,3478
	L_1-NRFPCA	72,1341	13,9995	4,1435	91,3698
	L_p-norm NRFPCA	73,2215	15,1123	4,7656	93,1128

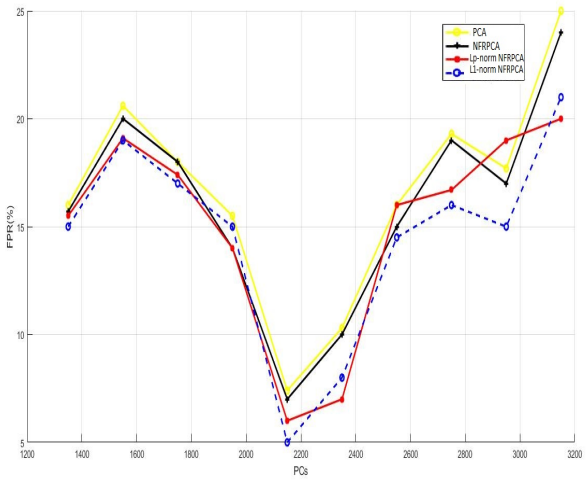

Figure 12: FPR vs. Training data using NSL-KDD dataset

The last figures (Figure.13 and Figure.14) exhibit the coorelation between CPU time and the number of principal components. As it is indicated clearly we observe that increasing the number of principal compenents PCs engender a huge consuming time. In addition to that, we can observe clearly in the figures that the suggested techniques are computationally speedy than the original algorithms.

To obtain higher precise results, we did an experiment in which we compared side by side the DR of every single category of attacks for PCA, NRFPCA, L_1-NFRPCA and L_p-norm NFRPCA. According to Table I and Table II. It is obviously clear that the DR of the

L_1-NRFPCA and L_p-norm NFRPCA for U2R and DOS attacks are often the highest compared to U2R and DOS attacks of NRFPCA and PCA.

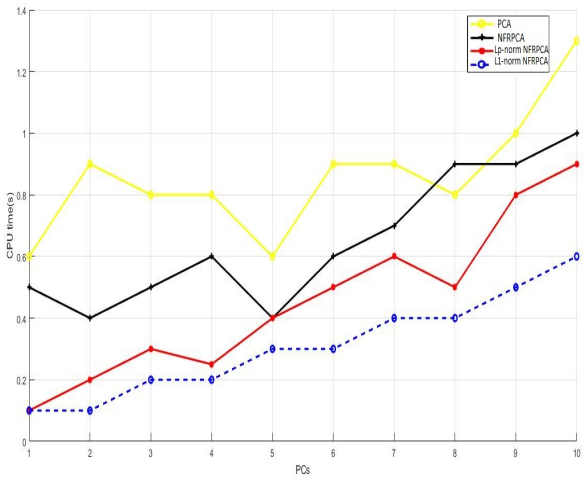

Figure 13: CPU time(s) vs. PCs using KDDCup99 dataset

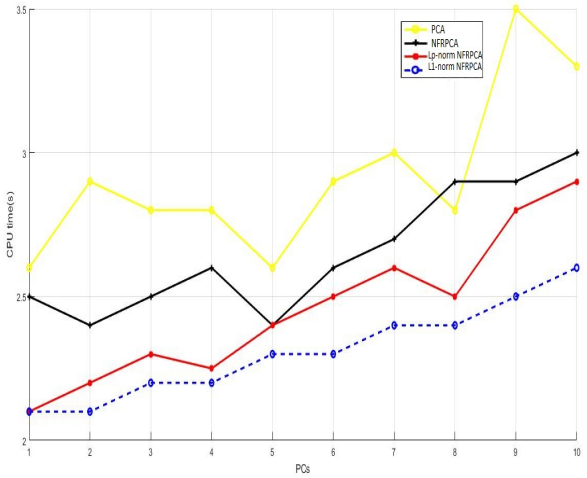

Figure 14: CPU time(s) vs. PCs using NSL-KDD dataset

7 Conclusion

As several linear statistical techniques Principal Component Analysis (PCA) has many shortcoming, it is limited just to Gaussian distribution and it has basically a high sensitivity to noise. In addition to that, principal components are frequently damaged by outliers, therefore feature extraction utilizing PCA are not credible if outliers exits in data. To tackle this issue, we proposed an effective new variants of nonlinear feature extraction techniques called Nonlinear Fuzzy Robust PCA for anomaly-based intrusion detection. The experiments performed on the popular databases (KDDcup99 and NSL-KDD), approved the effectiveness of the suggested approaches, the New variants outperfom NFRPCA and PCA in the detection of the most categories of attacks and in reducing the false positive alarms.

Conflict of Interest The authors declare no conflict of interest.

Acknowledgment This work is supported by CNRST-MOROCCO under the excellence program, grant no. 15UIT2016.

References

[1] E. Spafford and S. Kumar, *A software architecture to support misuse intrusion detection,* in Proceedings of the 18th National Information Security Conference, 1995, pp. 194-204.

[2] B. Caswell and J. Beale, *Snort 2.1 intrusion detection*. Syngress, 2004.

[3] J. P. A. Co, *Computer Security Threat Monitoring and Surveillance,* 1980.

[4] D. E. Denning, *Intrusion-Detection Model, IEEE Trans. Softw. Eng.*, vol. SE-13, no. 2, pp. 222-232, 1987.

[5] E. Popoola and A. O. Adewumi, *Efficient feature selection technique for network intrusion detection system using discrete differential evolution and decision.*," International Journal Network Security, vol. 19, no. 5, pp. 660-669, 2017.

[6] M. H. Aghdam and P. Kabiri, *Feature selection for intrusion detection system using ant colony optimization,*" International Journal Network Security, vol. 18, no. 3, pp. 420-432, 2016.

[7] A. S. Eesa, Z. Orman, and A. M. A. Brifcani, *A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems,*" Expert Systems with Applications, vol. 42,no. 5, pp. 2670-2679, 2015.

[8] A.Hadri, K.Chougdali and R.Touahni,*Intrusion detection system using PCA and Fuzzy PCA techniques*, in the proceeding of the Interntional Conference on Advanced Communication Systems and Information Security (ACOSIS), 17-19 October 2016, Marrakesh, Morocco.

[9] A.Hadri, K.Chougdali and R.Touahni, *Identifying intrusions in computer networks using Robust Fuzzy PCA*, in the proceeding of the IEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), 30 October-3 November 2017,Hammamet , Tunisia.

[10] Y. Bouzida and N. Cuppens-boulahia, *Efficient Intrusion Detection Using Principal Component Analysis. pp. 381-395. 2004*

[11] M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn, and L. Chang, *A novel anomaly detection scheme based on principal component classifier,* Miami Univ. Dept Electr. Comput. Eng. Tech. Rep, 2003.

[12] W. Wang and R. Batitti, *Identifying intrusions in computer networks with principal component analysis,* 2006, p. 8.

[13] L. Xu and A. L. Yuille, *Robust principal component analysis by self-organizing rules based on statistical physics approach,* IEEE Trans. on Neural Net., vol. 6, no. 1, pp. 131-143, 1995.

[14] R. L. Kashyap, M. J. Paulik, N. Loh, A. Automation, A. K. Jain, G. M. Jenkins, S. Francisco, and L. Sirovich, *Application of the Karhunen-Lokve Procedure for the Characterization of Human Faces,* vol. 12, no. 4, 1990.

[15] P. Luukka, *A New Nonlinear Fuzzy Robust PCA Algorithm and Similarity Classifier in Classification of Medical Data Sets,* Int. J. Fuzzy Syst., vol. 13, no. 3, pp. 153-162, 2011.

[16] Amal HADRI, Khalid Chougdali, and Raja Touahni."A Network Intrusion Detection based on Improved Nonlinear Fuzzy Robust PCA." 2018 IEEE 5th International Congress on Information Science and Technology (Cist).IEEE, 2018.

[17] M. Ringnr,*What is principal component analysis?,"* vol. 26, no. 3, pp. 303-304, 2008.

[18] J. Shlens, M. View, and I. Introduction, *A Tutorial on Principal Component Analysis,* 2014.

[19] T. N. Yang and S. D. Wang, *Robust algorithms for principal component analysis,* Pattern Recognit. Lett., vol. 20, pp. 927-933, 1999.

[20] L. Xu and A. L. Yuille, *Robust principal component analysis by self-organizing rules based on statistical physics approach,* IEEE Trans. Neural Net., vol. 6, no. 1, pp. 131-143, 1995.

[21] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, *A Detailed Analysis of the KDD CUP 99 Data Set,* no. Cisda, pp. 1-6, 2009.

[22] [Online] Available: http://kdd.ics.uci.edu/databases/kddcup99/task.html.

[23] [Online] Available: http://www.unb.ca/cic/research/datasets/nsl.html.