International Review on Computers and Software (I.RE.CO.S.), Vol. 11, N. 1
ISSN 1828-6003 *January 2016*

Network Intrusion Detection System Using PCA by Lp-Norm Maximization Based on Conjugate Gradient

Z. Elkhadir[1], K. Chougdali[2], M. Benattou[3]

Abstract – *Due to the fast growing of computer networks the potential for attacking those networks also became important. Therefore, all enterprises should implement various systems that supervise their network infrastructure security. To detect any eventual attacks, many Intrusion Detection Systems (IDSs) have been used in recent years. However, the most of them operate more often on enormous network traffic data with multiple redundant features. As a result, the IDS generates a high false alarms rate, which makes the intrusion detection inefficient and imprecise. To overcome that, several techniques for data dimensionality reduction have been proposed, such as Principal Component Analysis (PCA).*
Nonetheless, the classical PCA approach that is based on the L2-norm maximization is very sensitive to outliers. As a solution to this weakness, we propose to introduce a new variant of PCA called PCA Lp-norm using conjugate gradient algorithm to solve the Lp-norm optimization problem. The main idea behind this new method relies on the Lp-norm, which is more robust to the presence of outliers in data. Extensive experiments on two well-known datasets namely KDDcup99 and NSL-KDD prove the effectiveness of the proposed approach in terms of network attacks detection, false alarms reduction and CPU time minimization. ***Copyright © 2016 Praise Worthy Prize S.r.l. - All rights reserved.***

Keywords*: IDS, PCA, Conjugate Gradient, NSL-KDD, Kddcup99*

Nomenclature

X	The training data matrix
d	Dimension of data
N	Number of samples in training data
m	Number of principal components
W	Matrix of projection vectors
S	Covariance or scatter matrix of X
∇_w	Gradient vector
α	Learning rate
$s(.)$	Sign function equals 1 or -1 or 0.
$\delta(.)$	Dirac delta function
β	Scalar used in conjugate gradient to determine the next direction
δ	Small random vector
ε	Very small number

I. Introduction

An intrusion can be defined as any intentional action attempting to break a computer system by exploiting its vulnerabilities or theft a personal information.

The automated process that can help to detect intrusion is called Intrusion Detection System (IDS).

There are two types of IDS, Host-based Intrusion Detection System (HIDS) and Network-based Intrusion Detection System (NIDS).

The first type ensures the protection of a certain operating system, while NIDS takes the responsibility of monitoring computer network. The intrusion detection techniques are grouped in two common categories, misuse-based detection and anomaly-based detection.

The concept behind the first one is based on recognizing intrusion by comparing it with attacks, which are already stored in a database of attacks signatures. This detection technique is commonly used in STAT [1] and Snort [2]. On the other side, anomaly detection is conducted in two steps: learning phase and detection phase. In the first step, we build a profile or a model of the normal system behavior. While in the second step, the actual system behavior is compared with ones in the normal system. If the IDS detects any deviation from normal system behavior, it will generate an alert for possible intrusionand notify the network administrator. The anomaly based approach has been developed firstly by Anderson [3] and Denning [4], and then implemented in some IDS like IDES [5] and EMERALD [6]. Unfortunately, this approach has some weaknesses, beacause it has a modest detection rate and the produced false alarm rate is high.

To deal with this limitation and thereafter enhancing the performance of an network IDS,various techniques were recently suggested [7]-[9].

Un addition to these techniques,many data dimensionality reduction algorithms have been also exploited to improve IDS performance.

DOI: 10.15866/irecos.v11i1.8370

Among them, the popular one is Principal Component Analysis (PCA) used previously in [10]-[12] to developp an IDS. This method focuses on searching a set of projection vectors that reduces dimensionality of original data such that the variance of these projected data is maximized.

The obtained low-dimensional linear subspace is spared as much as possible from the useless information. In addition it preserves the data structure of the original space. However, the performance of the traditional PCA is negatively affected by the existence of outliers, which are defined as data points that deviate widely from the rest of the data. This fact comes from the use of the large Euclidean norm (L2-norm) in the classical PCA which leads to inaccurate detection results. To handle this problem, several robust PCA variants have been developed [13]-[15].

In this paper, we propose firstly to use PCA Lp-norm [16]in context of intrusion detection and compare it to the classical PCA, secondly we improve this method by introducing an iterative algorithm based on conjugate gradient. The remainder of this paper is organized as follows. In Section II, we formulate mathematically the original problem of L2-norm PCA, after that, we present the Lp-norm based optimization problem.

The Section III is dedicated to explain the new solution which exploit the conjugate gradient algorithm. In Section IV, we discuss the two datasets on which we have conducted our experiments.Section V exposes the experimental results, which demonstrate the effectiveness of the new PCA Lp and illustrates its advantage over the old PCA Lp-norm. Finally, section VI summaries the principal obtained results and concludes the paper with a discussion about future research directions.

II. Mathematical Formulation of the Problem

Given the data matrix $X = \left[x_1,\ldots,x_N\right]$, $x_i \in R^d$, where N is the total number of data samples.

We assume that all the samples $\left\{x_i\right\}_{i=1}^{N}$ have zero mean. The main goal of the classical L2-norm PCA is to find a linear tranfsormation $W : R^d \to R^m$, where $W = \left[w_1,...,w_m\right] \in R^{d\times m}$ and $W^T W = I_m$. I_m is the $m\times m$ identity matrix. Then the original high-dimensional data x_i is tranformed to a low-dimensional vector $W^T x_i$. In order to find the projection matrix W, the following objective function is maximized:

$$F_2\left(W\right) = \frac{1}{2}\sum_{i=1}^{N}\left\|W^T x_i\right\|_2^2 = \frac{1}{2}tr\left(W^T SW\right) \qquad (1)$$

where, $S = XX^T$ represents the total scatter matrix of X, tr(.) is the trace operator and $\|.\|_2$ denote the L2-norm of a vector.

The key step to solve the problem (1) is to solve the eigenvalue problem $SW = W\Lambda$ where Λ is a diagonal matrix containing eigenvalues of S.

Note that the total scatter in (1) is dominated by samples with large normsand thus the outliers can be amplified by using the L2-norm.

This fact makes the process of finding the orthonormal projection vectors inaccurate. In order to deal with this issue and increase the robustness to the outlier samples, we propose to maximize the Lp-norm instead of the Euclidean norm.

Then the problem (1) becomes [16]:

$$F_p\left(W\right) = \frac{1}{p}\sum_{i=1}^{N}\left\|W^T x_i\right\|_p^p = \frac{1}{p}\sum_{i=1}^{N}\sum_{j=1}^{m}\left|w_j^T x_i\right|^p \qquad (2)$$

Here, $\|.\|_p$ denote the Lp-norm of a vector. Keep in mind that the objective function (2) is identical to (1) when $p = 2$.

On the other hand, directly solving the problem (2) is difficult, therefore we can use a greedy procedure to reach the projection vectors $\left\{w_1,...,w_m\right\}$ one by one.

II.1. Solution for (m=1)

To make the problem in (2) more manageable, we firstly search just one projection vector (m=1) then (2) becomes:

$$w^* = \underset{w}{arg\,max}\,F_p(w) = \underset{w}{arg\,max}\,\frac{1}{p}\sum_{i=1}^{N}\left|w^T x_i\right|^p \qquad (3)$$

It has been known that the solution of this optimization problem can be reached by.

We solve this optimization problem by takingthe partial derivative of $F_p\left(w\right)$ with respect to w.

Nevertheless, we have to deal with the absolute value operation by introducing a sign function as follows:

$$s\left(a\right) = \begin{cases} 1 & a > 0 \\ 0 & a = 0 \\ -1 & a < 0 \end{cases} \qquad (4)$$

This allows us to rewrite $F_p\left(w\right)$ in (3) as:

$$F_p\left(w\right) = \frac{1}{p}\sum_{i=1}^{N}\left[s\left(a_i\right)a_i\right]^p, \qquad a_i = w^T x_i \qquad (5)$$

Computing the partial derivative of $F_p\left(w\right)$ with respect to w, we obtain:

$$\begin{aligned}\underset{w}{\nabla} &= \frac{dF_p(w)}{dw} = \sum_{i=1}^{N} \frac{dF_p(w)}{da_i}\frac{da_i}{dw} = \\ &= \sum_{i=1}^{N} \left[s(a_i) a_i \right]^{p-1} \left[s'(a_i) a_i + s(a_i) \right] \\ &= \sum_{i=1}^{N} s'(a_i) s^{p-1}(a_i) a_i^p x_i + \sum_{i=1}^{N} s^p(a_i) a_i^{p-1} x_i \\ &= 2\sum_{i=1}^{N} \delta(a_i) s^{p-1}(a_i) a_i^p x_i + \sum_{i=1}^{N} s(a_i) |a_i|^{p-1} x_i \end{aligned} \tag{6}$$

where $\delta(\ )$ is the Dirac delta function. The first term equals zero if $a_i \neq 0$ for all x_i then we obtain:

$$\underset{w}{\nabla} = \sum_{i=1}^{N} s\left(w^T x_i\right) \left|w^T x_i\right|^{p-1} x_i \tag{7}$$

A particular case takes place when $p \leq 1$. Effectively here, the partial derivative $\underset{w}{\nabla}$ is not well defined for some vectors w if there exist some x_i's such that $w^T x_i = 0$. However, we can skip this singular case by adding a small random vector δ toeach w. Note that we can do such a manipulation just if the number of samples is limited.

So, the steepest gradient method can beused to get the solution of the problem (3). The entire optimization procedure is represented as PCA-Lp(G) [16] in the following steps:

1. Initialization: $t \longleftarrow 0$. Set $w(0)$ such that $\|w(0)\|_2 = 1$
2. Singularity check (applies only if $p \leq 1$)
   If $\exists i$, such that $w^T(t) x_i = 0$, $w(t) \longleftarrow (w(t)+\delta) / \|w(t)+\delta\|_2$. Here, δ is a small random vector.
3. Computation of gradient ∇_w using (7).
4. Gradient search: $w(t+1) \longleftarrow w(t) + \alpha \nabla_w$, where α is the learning rate.
5. Normalization:

- $t \longleftarrow t+1$
- $w(t) \longleftarrow \dfrac{w(t)}{\|w(t)\|_2}$

6. Convergence check.

- If $\|w(t) - w(t-1)\|_2 > \varepsilon$ go to Step 2.

  Else, $w^* \longleftarrow w(t)$. Stop iteration.

Note that, the steepest gradient method finds a big difficulty to converge if the learning rate α is high, whereas with the small values of α the convergence is not fast enough.

In this paper, the parameter α is set to $\dfrac{10}{N}$, where N is the total number of training samples.

II.2. *Multiple Feature Extraction $(m > 1)$*

In this subsection, the PCA Lp-norm algorithm is generalized to extract multiple features $(m > 1)$.

The proposed method is applied and it procedure is as follows [16]:

1. Let $w_0 = 0$ and $X_0 = X$.
2. For i = 1..*m*
   - Set $X_i = \left(I_d - w_{i-1} w_{i-1}^T\right) X_{i-1}$
   - Apply PCA Lp-norm On X_i.
3. Output $W^* = [w_1, \ldots, w_m]$

One of major weaknesses of the steepest gradient (SG) is its slowness.

This fact comes from the line search strategy that is implemented in this method.

Hence, the convergence of the steepest gradient algorithm is hard to reach, which will causes a slow maximization of any given function.

III. The Proposed Solution

To deal with the steepest gradient issue, we propose to use the conjugate gradient (CG). The most important advantages of this method are the low memory requirements and the convergence speed.

Having said that, our algorithm will be inspired by[17] and looks like steepest gradient procedure.The following steps gives more details to explain our approach:

1. Initialization: $t \longleftarrow 0$. Set $w(0)$ such that $\|w(0)\|_2 = 1$, $d(-1) = 0$, $b = 1$, here d is direction vector.
2. Singularity check (applies only if $p \leq 1$)
   - If $\exists i$, such that $w^T(t) x_i = 0$, $w(t) \longleftarrow (w(t)+\delta) / \|w(t)+\delta\|_2$. Here, δ is a small random vector.
3. Computation of gradient ∇_w using (7).
4. Computation of β: $\beta = \dfrac{\|\nabla_w\|_2^2}{b}$
5. Compute direction vector: $d(t) \longleftarrow \nabla_w + \beta \cdot d(t-1)$
6. Update: $w(t) \longleftarrow w(t-1) + \alpha \cdot d(t)$, where α is the learning rate.
7. $b = \|\nabla_w\|_2^2$
8. Normalization.

   $t \longleftarrow t+1$

$$w(t) \longleftarrow \frac{w(t)}{\|w(t)\|_2}$$

9. Convergence check.
   - If $\|w(t)-w(t-1)\|_2 > \varepsilon$ go to Step 2.

Else, $w^* \longleftarrow w(t)$. Stop iteration.

As we can see, two critical parameters are required in the maximization process. The first one is β and the second one is called the direction search $d(t)$. In order to get it, we need only to know three vectors: the current, the previous gradients and the previous direction $d(t-1)$. Every update of the projection vector further maximizes $F_p(W)$, consequently choosing an initial $w(0)$ becomes definitely critical. In our analysis, we set the initial vector $w(0)$ at the solution of the classical PCA. But, we can try also other techniques like re-execute the CG algorithm many times with various initial value $w(0)$ and picking the best one.

IV. The Simulated Databases

IV.1. KDDcup99

The principal task for the KDDcup99 classifier learning contest [18] is to present a predictive model able to recognize legitimate (normal) and illegitimate (called intrusion or attacks) connections in a computer network.

The entire training dataset contained about 5,000,000 connection records. In this paper we work with the training 10% dataset consisted of 494,021 records which contain 97,278 normal connections (i.e. 19.69%). Each connection record is composed of 41 different attributes that describe the different features of the corresponding connection, and the value of the connection is labeled either as an attack with one specific attack type, or as normal. Each attack type falls exactly into one of the following four categories:

1. Probing: surveillance and other probing, e.g., port scanning;
2. DOS: denial-of-service, e.g. syn flooding;
3. U2R: unauthorized access to local superuser (root) privileges, e.g., various buffer overflow attacks;
4. R2L: unauthorized access from a remote machine, e.g. password guessing

The test dataset used in our simulation experimentsis composed of 311,029 connections. It is important to note that the test dataincludes some specific attack types which does not exist in the training data.

IV.2. NSL-KDD

This data set suggested to solve some of the inherent problems of the KDD'99 data set [19], and has the following advantages over the original KDD[20]:

1. The train set does not contain redundant records. so the classifiers will not be affected by more frequent records.
2. The proposed test set does not include duplicate records; consequently, the performance of the learners are not sensetive to the techniques which give good detection rates with frequent records.
3. The number of selected records from each difficulty level group is inversely proportional to the percentage of records in the original KDD data set. As a result, the classification rates of distinct machine learning methods vary in a wider range, which makes it more efficient to have an accurate evaluation of different learning techniques.
4. The number of records in the train and test sets are reasonable, which makes it affordable to run the experiments on the complete set without the need to randomly select a small portion. Consequently, evaluation results of different research works will be consistent and comparable.

IV.3. Transformation Process

The datasets are constructed by discrete and continuous attributes values. We have applied the transformation concept used in [10] to convert the discrete attributes values to continuous values. Let's have a look on this process:

Suppose a discrete attribute i contains k values. we correspond to i the k coordinates consisting of one and zeros, then, we will obtain one coordinate for every possible value of the attribute. As an example, if we focus on the protocol type attribute which can be tcp, udp or icmp. According to the transformation concept, this attribute will be converted to three coordinates, as a consequence, suppose a connection record contains a tcp (resp. udp or icmp) then the corresponding coordinates will be (1,0,0) (resp. (0,1,0) or (0,0,1)).

V. Experiments and Discussion

In this section, we firstly present and discuss the various results obtained using PCA Lp-norm which exploits steepest gradient (SG). Secondly, we test our optimized approach which is based on gradient conjugate (CG). The performance of an IDS is evaluated by its ability to make correct predictions. Depending on the real nature of a given event compared to the prediction from the IDS, we can usethe following measures to evaluate the performance of IDS:

$$DR = \frac{TP}{TP+FN} \times 100 \tag{8}$$

$$FPR = \frac{FP}{FP+TN} \times 100 \tag{9}$$

where true positives (TP) correspond to intrusions correctly predicted.

False negatives (FN) refer to intrusions wrongly classified; false positive (FP) are normal instances wrongly classified, and true negatives (TN) are normal instances successfully predicted.

Hence, based on these performance indicators, an efficient IDS should have a high Detection Rate (DR) and a low False Positive Rate (FPR). As a first experiment, as regards KDDcup99, we have chosen a number of training samples composed of 1000 normal, 100 DOS, 50 U2R, 100 R2L, and 100 PROBE all randomly selected from the 10% of training dataset.

A test samples composed of 100 normal data, 100 DOS data, 50 U2R data, 100 R2L data, and 100 PROBE, selected from the test dataset. For NSL-KDD, the simulation settings are the same as those used in KDDcup99. After applying PCA Lp-norm and classical PCA on training samples, we have obtained the principal components (PC).

The number of PC determines the dimension of the new reduced samples. Then, we project the test samples on the subspace spanned by these principal components, varying their numbers.

The objective of this experiment is to search the optimal number of PCs which contribute significantly in increasing detection rate (DR) and decreasing FPR.

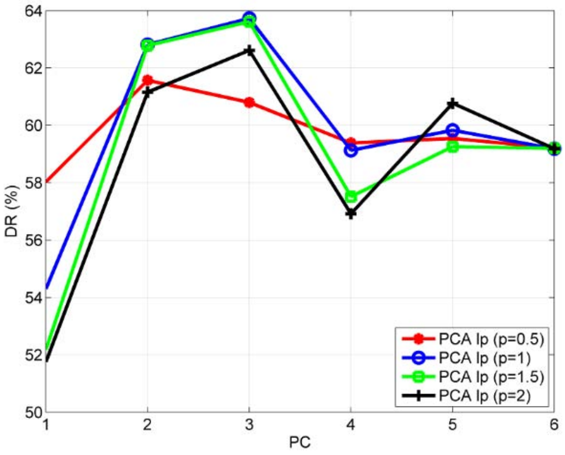


Fig. 1. Detection rate's comparison of PCA Lp for KDDcup99

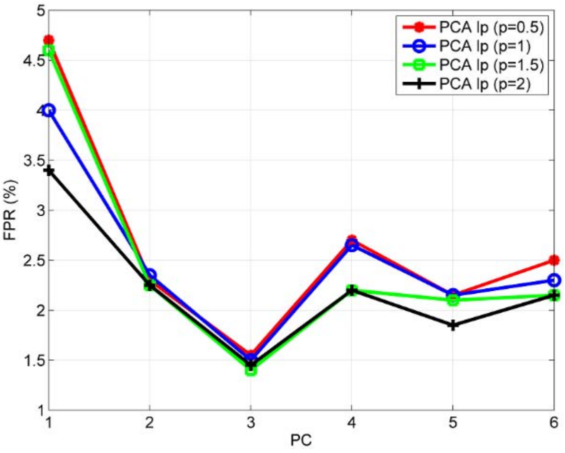


Fig. 2. False positive's comparison of PCA Lp-norm for KDDcup99

As shown in Fig. 1 and Fig. 2, the KDDcup99's simulation results, we can deduce clearly that three principal components (PC) is exactly the number we are looking for.

So using these PCs, we observe that DR reaches his peak. In the other side, the FPR is minimized.

Similarly, for the second dataset (Fig. 3 and Fig. 4), the same interpretations still valid, except for (p=1), where four PCs are needed to get optimal results.

Hence, we can conclude that the PCA Lp-norm is much better than classical PCA specially when p takes 1 and 1.5.

In order to go deeper in our investigation we have calculated the detection rate of every type of attack (DOS, U2R, R2L, PROBE).

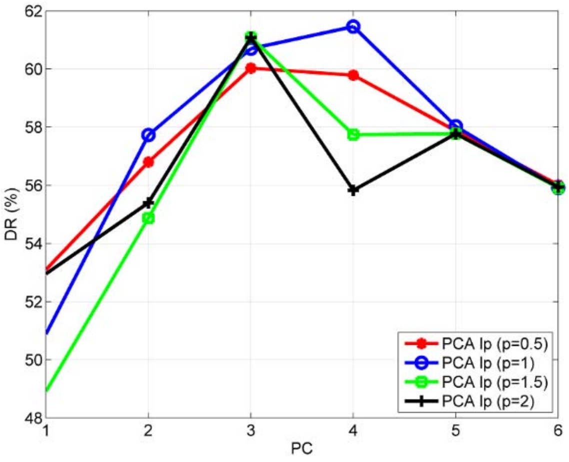


Fig. 3. Detection rate's comparison of PCA Lp-norm for NSL-KDD

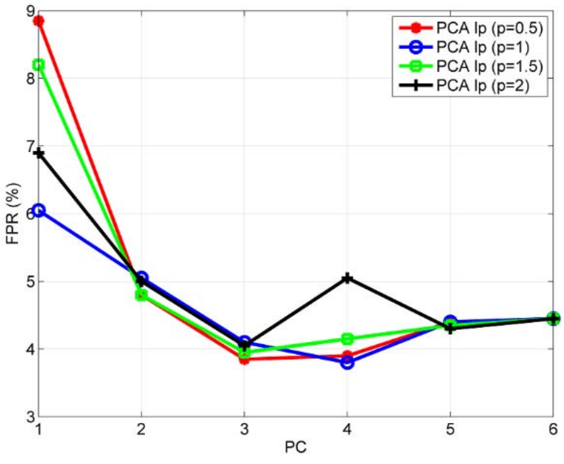


Fig. 4. False positive'scomparison of PCA Lp-norm for NSL-KDD

TABLE I
ATTACK'S DETECTION RATE (%) OF PCA LP

Database	The method	DOS	U2R	R2L	Probe
KDDcup99	PCA L1	93.65	11.2	4.2	75.5
	PCA L1.5	93.15	11.1	4.2	70.05
	PCA L2	92.85	11	4.2	65.65
NSL-KDD	PCA L1	74.6	11.5	15.8	56.3
	PCA L1.5	73.4	11.4	15.8	53.4
	PCA L2	72.65	11.3	15.8	54.35

According to Table I, it is shown that the IDS which uses PCA Lp-norm detects more efficiently DOS and PROBE attacks for the two datasets. The other types of attacks (U2R and R2L) are similarly recognized with a low detection rate and for all values of *p*.

This can be explained by the fact that KDDcup99 and NSL-KDD are containing a few connection records for these type of attacks. To generalize our conclusions about the previous experiment, we varied the number of connection records in the training datasets.

According to Table II and Table III, we note that DOS and PROBE attacks still well detected by the IDS. However, the other truth concerning identification U2R and R2L attacks persists. The obtained results confirm that PCA Lp-norm has good performance from the point of view of a better attacks detection and lower false alarms. Unfortunately, we have noted that the CPU time consumed by PCA Lp-norm which is based on steepest gradient (SG) can be seen as its major weakness. To optimize that, we propose to employ a new PCA Lp which relies on conjugate gradient (CG). Hereafter, we distinguish the two PCA Lp-norm by using the notations: (SG Lp) for the old solution, and (CG Lp) for the proposed one.

As illustrated in Fig. 5 and Fig. 6, two figures that represent the results which flow from using KDDcup99, we deduce that using a direction vectorinformation rather than the gradient vector keeps the detection rate almost intact. However, it leads to a faster performance.

As an additional details, Table IV shows us that the detection of individual attacks for $p = 1$ still almost similar. Except those of DOS and PROBE, where we observe that from a certain number of training data (9150) the conjugate gradient surpassesslightly the other optimisation technique.

TABLE II
TRAINING DATA VS. ATTACK'S DETECTION (%) FOR KDDCUP99

Training data	PCA Lp	DOS	U2R	R2L	Probe	time (s)
2650	p=1	**91.05**	**12.9**	**4.35**	**76.8**	1.05
	p=1.5	88.50	12.7	4.3	69.80	0.99
	p=2	89.05	12.6	4.3	64	0.06
3950	p=1	**93.7**	**12.1**	**4.2**	**77.15**	1.79
	p=1.5	89.75	11.8	4.15	69.1	1.40
	p=2	91.2	11.8	4.15	64.9	0.07
5250	p=1	**92.2**	**10.8**	**4.45**	**80**	2.20
	p=1.5	89.3	10.8	4.45	71.6	1.62
	p=2	89.2	10.8	4.45	65.35	0.09
6550	p=1	**92.95**	**9**	**3.8**	**80.3**	3.17
	p=1.5	89.75	9	3.75	75.5	2.77
	p=2	90.20	9	3.75	70.45	0.15
7850	p=1	**92.45**	**10.9**	**5.45**	**78.85**	3.76
	p=1.5	90.05	10.4	5.45	72.35	2.72
	p=2	90.15	10.3	5.45	66.95	0.15
9150	p=1	**91.95**	**9.9**	**5**	**79.55**	4.49
	p=1.5	88.25	9.7	4.95	70.75	3.67
	p=2	89.15	9.6	4.95	65.70	0.17
10450	p=1	**92.65**	**9.8**	**5.3**	**78.8**	5.02
	p=1.5	90.5	9.7	5.25	71.25	4.19
	p=2	91.10	9.5	5.2	65.4	0.19
11750	p=1	**94.3**	**9.8**	**4.8**	**79.8**	5.55
	p=1.5	93.15	9.3	4.75	73.6	4.60
	p=2	92.25	9.2	4.75	67.05	0.21

TABLE III
TRAINING DATA VS. ATTACK'S DETECTION (%) FOR NSL-KDD

Training data	PCA Lp	DOS	U2R	R2L	Probe	time (s)
2650	p=1	**68**	**10**	**14.21**4.2	**58.6**	0.93
	p=1.5	67.2	9.5	14.1	52.15	0.78
	p=2	66.35	9.5		52.95	0.05
3950	p=1	**67.85**	**11.1**	**17.7**	**58.75**	1.51
	p=1.5	67.25	11.1	17.5	54.65	1.13
	p=2	66.20	11.1	17.5	53.95	0.07
5250	p=1	**68.3**	**11**	**17.7**	**60.9**	1.88
	p=1.5	68.1	10.7	17.4	58.1	1.4
	p=2	89.2	10.8	4.45	65.35	0.09
6550	p=1	69.65	**9.7**	**23.2**	**62.7**	2.46
	p=1.5	**69.75**	9.4	**23.2**	56.85	1.8
	p=2	68.55	9.4	23.1	55.25	0.13
7850	p=1	70.4	**9.4**	**19**	**63.15**	2.73
	p=1.5	**70.7**	9.1	18.1	59	1.8
	p=2	69.05	9	18.1	57.55	0.15
9150	p=1	**69.55**	**8.8**	**20.8**	**60.7**	3.75
	p=1.5	68.6	8.7	20.7	54.55	2.69
	p=2	67.4	8.4	20.7	53.05	0.2
10450	p=1	67.75	**7.3**	**24.7**	**63.2**	4.32
	p=1.5	**68.04**	7.3	24.1	58.9	2.42
	p=2	67.05	7.3	24.1	56.2	0.21
11750	p=1	67.5	**9.7**	**25**	**61**	4.40
	p=1.5	**68.3**	9.7	24.6	57	2.79
	p=2	66.7	9.6	24.6	55.9	0.23

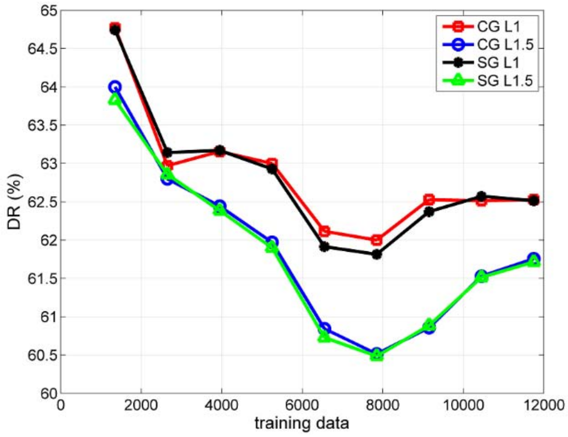


Fig. 5. Training data vs. detection rate for KDDcup99

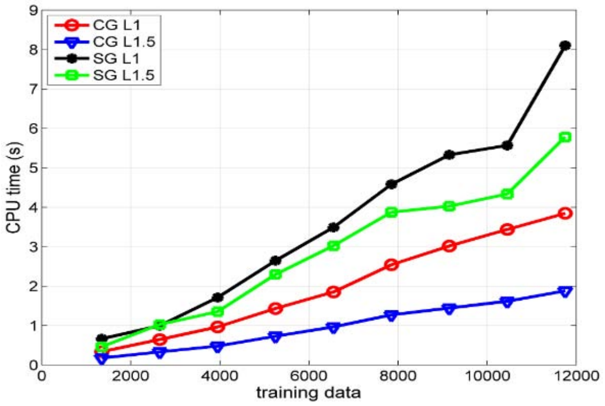


Fig. 6. Training data vs. consumed CPU time

For NSL-KDD, as mentioned in Fig. 7 and Fig. 8, we don't notice any improvement in term of general attack's identification compared to SG Lp. Nonetheless CG Lp consumes less CPU time.

TABLE IV
TRAINING DATA VS. ATTACK'S DETECTION (%) FOR KDDCUP99

Data	Gradient Type	DOS	U2R	R2L	Probe
2650	CG L1	**94.10**	**13.1**	**4**	**72.8**
	SG L1	94.10	13.		73.05
3950	CG L1	**93.05**	**10.9**	**5**	**75.95**
	SG L1	93.25	10.9	5	77.9
5250	CG L1	**93.6**	**10.9**	**5**	**78.15**
	SG L1	92.9	10.9	5	79.5
6550	CG L1	**92.95**	**9.8**	**3.8**	**77.95**
	SG L1	92.95	9.8	3.8	78
7850	CG L1	**93.1**	**11.8**	**4.55**	**78.1**
	SG L1	92.55	11.8	4.55	77.95
9150	CG L1	**92.1**	**9.6**	**4.9**	**78.5**
	SG L1	91.45	9.6	4.9	78.2
10450	CG L1	**93.55**	**11.5**	**5.55**	**80.4**
	SG L1	92.85	11.5	5.55	80.4
11750	CG L1	**93.05**	**8.6**	**4.45**	**78.7**
	SG L1	93.10	8.6	4.75	78.6

TABLE V
TRAINING DATA VS. ATTACK'S DETECTION (%) FOR NSL-KDD

Data	Gradient Type	DOS	U2R	R2L	Probe
2650	CG L1.5	**69.65**	**11.4**	**13.2**	**54.35**
	SG L1.5	69.60	11.4	13.2	54.25
3950	CG L1.5	**68.9**	**10.3**	**13.4**	**52.9**
	SG L1.5	68.8	10.3	13.4	52.2
5250	CG L1.5	**68.6**	**10.4**	**20.2**	**52.85**
	SG L1.5	68.55	10.4	20.2	52.45
6550	CG L1.5	**69.25**	**9.3**	**17.2**	**54.95**
	SG L1.5	69.2	9.3	17.2	54.9
7850	CG L1.5	**68**	**10**	**20.4**	**58.25**
	SG L1.5	67.9	10	20.4	57.75
9150	CG L1.5	**69.05**	**9.4**	**22.2**	**57.8**
	SG L1.5	69.35	9.4	22.2	58.05
10450	CG L1.5	**67.85**	**9.2**	**21.4**	**58.9**
	SG L1.5	68.15	9.2	21.4	58.65
11750	CG L1.5	**68.6**	**8.4**	**23.5**	**59.7**
	SG L1.5	68.8	8.4	23.5	59.4

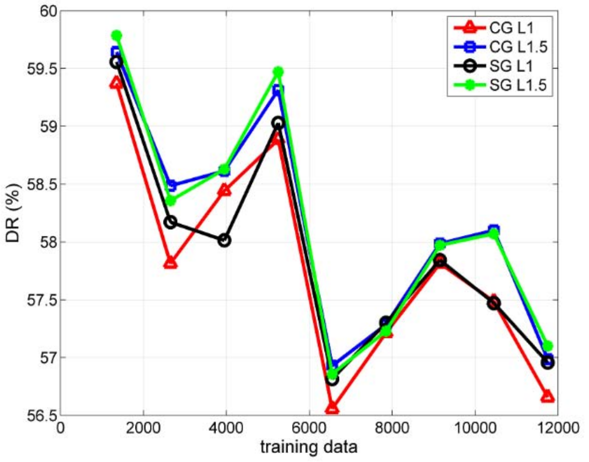


Fig. 7. Training data vs. DR for NSL-KDD

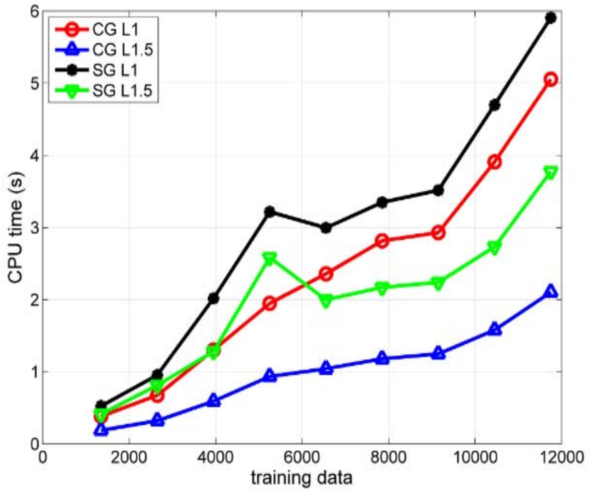


Fig. 8. Training data vs. CPU time

Moreover, concerning detection of individual attacks,and comparaing CG and SG performanceswith $p = 1.5$, Table V tells us that the proposed approach overcomes slightly the old PCA Lp-norm in DOS and PROBE identification.

VI. Conclusion

The classical principal component analysis (PCA) is paralyzed by its intrinsic sensitivity to outliers when it is applied to intrusion detection. As a solution to handle this weakness, we have proposed a new PCA Lp-norm based on conjugate gradient.

This technique tries to find projections that maximize total covariance using Lp-norm instead of L2-norm. Experimental results showed that, our approach takes advantage over PCA Lp-norm which relies on steepest gradient in attacks detection and consuming less CPU time. However our approach still not enough speedy due to it greediness nature.

The future work could aim at the deeper study of the algorithm, making it more rapid and more accurate with the adoption of a non greedy approach.Furthermore, the way of choosing the learning rate *a* in this paper is kind of trivial, so it is worthy to look for a more reasonable way to determine it.

References

[1] Kumar, S., Spafford, E., *A software architecture to support misuse intrusion detection*, Proceedings of the 18th National Information Security Conference (Pages: 194-204 Year of Publication: 1995).

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

[3] J. P. Anderson, Computer security threat monitoring and surveillance, Fort Washington, Pennsylvania, Tech. Rep., 1980.

[4] D. E. Denning, An intrusion-detection model, *IEEE Transactions on Software Engineering*, n. 2, pp. 222-232, 1987.

[5] T. Lunt, A. Tamaru, F. Gilham, R. Jagannathan, C. Jalali, H. Javitz, A. Valdes, P. Neumann, and T. Garvey, A real-time intrusion-detection expert system (ides), Computer Science Laboratory, SRI International, Menlo Park, California,, Tech. Rep., 1992.

[6] Porras, P. A., Neumann, P. G., *Emerald: Event monitoring enabling response to anomalous live disturbances*, Proceedings of the 20th national information systems security conference (Pages: 353-365 Year of Publication: 1997).

[7] Mohamed Mubarak, T., Sajitha, M., Appa Rao, G., Sattar, S., Secure and Energy Efficient Intrusion Detection in 3D WSN, (2014) *International Journal on Information Technology (IREIT),* 2 (2), pp. 48-55.

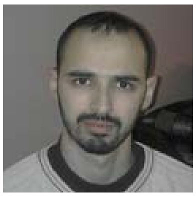
[8] Mohamed Mubarak, T., Appa Rao, G., Sattar, S.A., Sajitha, M.,

Efficient intrusion detection ensuring connectivity in 2D and 3D WSN, (2014) *International Review on Computers and Software (IRECOS)*, 9 (2), pp. 219-229.
[9] Deepa, A.J., Kavitha, V., Neurofuzzy and genetic network programming based intrusion detection system, (2014) *International Review on Computers and Software (IRECOS)*, 9 (2), pp. 295-301.
[10] Bouzida, Y., Cuppens, N., Cuppens-Boulahia, N., Gombault, S., Efficient intrusion detection using principal component analysis, *3eme conference sur la Sécuritéet Architectures Réseaux (SAR)* (Year of Publication: 2004).
[11] M.-L. Shyu, S.-C. Chen, K. Sarinnapakorn, and L. Chang, *A novel anomaly detection scheme based on principal component classifier*, Miami University, Dept of electrical and computer engineering, Tech. Rep., 2003.
[12] Wang, W., Battiti, R., Identifying intrusions in computer networks with principal component analysis, *Proceedings of The First International Conference on Availability, Reliability and Security, ARES* (Pages: 8-pp Year of Publication: 2006).
[13] H. Xu, C. Caramanis, and S. Mannor, Outlier-robust pca: The highdimensional case, *IEEE Transactions on Information Theory*, vol. 59, no. 1, pp. 546–572, 2013.
[14] C. Pascoal, M. Oliveira, A. Pacheco, and R. Valadas, Detection of outliers using robust principal component analysis: A simulation study, *Combining Soft Computing and Statistical Methods in Data Analysis*. Springer, pp. 499–507, 2010
[15] N. Kwak, Principal component analysis based on l1-norm maximization, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 9, pp. 1672–1680, 2008.
[16] N. Kwak, Principal component analysis by l*p*-norm maximization, *IEEE Transactions on, Cybernetics*, vol. 44, no. 5, pp. 594–609, May 2014.
[17] R. Fletcher, C. M. Reeves, Function minimization by conjugate gradients, *The computer journal*, vol. 7, no. 2, pp. 149–154, 1964.
[18] KDD database web site http://kdd.ics.uci.edu/databases/kddcup99/
[19] NSL-KDD web site : http://nsl.cs.unb.ca/NSL-KDD/
[20] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, ”A Detailed Analysis of the KDD CUP 99 Data Set” In *Proceeding of the 2009 IEEE symposium on computational Intelligence in security and defense application (CISDA)*, 2009.

Authors' information

[1,3]LASTID laboratory, Faculty of Science, Ibntofail University, Kenitra, Morocco.

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



Zyad Elkhadir is a PhD student in Faculty of science, IbnTofail University, Kenitra, Morocco. He obtained his Master degree in computer science in 2013 from the same Faculty. His main research interest is to develop new feature extraction algorithms for pattern recognition problem such as network intrusion detection.

Khalid Chougdali is an associate Professor of Computer Science at the National School of Applied Sciences, Kénitra. In 2010 he obtained his PhD degree from Mohamed V-Agdal university in computer science. His main research interest are network security, pattern recognition and biometrics.
E-mail: chougdali@yahoo.fr

Mohamed Benattou is a Professor of Computer Science at the IBN TOFAIL University – KÉNITRA where he directs the Computer Science and Telecommunication Laboratory. He has also held several positions in her French academic career: University of PAU, University of ORSAY Paris XI, 3IL and Xlim Laboratory. His research interests include distributed testing, secure testing, and software testing.