Network Intrusion Detection System Using L1-norm PCA

CHOUGDALI Khalid
GEST Research group
National School of Applied Sciences (ENSA)
Ibn Tofail University, Kenitra
Email: chougdali@gmail.com

ELKHADIR Zyad
RLCST Research Laboratory
Ibn Tofail University, Kenitra
Email: zyad.elkhadir@gmail.com

BENATTOU Mohammed
RLCST Research Laboratory
Ibn Tofail University, Kenitra
Email: mbenattou@yahoo.fr

***Abstract*—The rapid evolution of information and communication technologies leads to a big networks security problem. For this reason, the Intrusion Detection System (IDS) has been developed in order to detect and prevent computer network attacks. However, the majority of IDSs operate on huge network traffic data with many useless and redundant features. Consequently, the IDS generates a lot of false alarms and the intrusion detection process becomes difficult and imprecise. To improve the performance of an IDS, many data dimensionality reduction methods, such as Principal Component Analysis (PCA), have been proposed. However, the classical PCA approach, that is based on the covariance matrix of the data, is very sensitive to outliers. In order to overcome this problem, we propose to introduce a new variant of PCA namely L1-norm PCA. This new method is based on the L1-norm maximization, which is more robust to outliers, instead of the Euclidean norm in the classical PCA. Extensive experiments on the well-known KDDcup99 dataset are exploited for testing the effectiveness of the proposed approach. Obtained results confirm the superiority of L1-norm PCA over the traditional PCA in terms of network attacks detection and false alarms reduction.**

I. Introduction

An intrusion can be seen as any deliberate action that tries to manipulate a personal information or break a computer system by exploiting its vulnerabilities. The process which attempts to detect automatically intrusion is called Intrusion Detection System (IDS). There are two types of IDSs, Host-based Intrusion Detection System (HIDS) and Network-based Intrusion Detection System (NIDS). A HIDS ensures the protection of a certain operating system while a NIDS protects a computer network. The intrusion detection techniques fall into two common categories, misuse-based detection and anomaly-based detection. For the first one, the idea behind it is to recognize intrusion by comparing it with attacks already stored in a database of attacks signatures. Examples of misuse detection techniques are STAT [1] and Snort [2].

On the other hand, anomaly detection has two phases: learning phase and detection phase. In the learning phase, we construct a profile or a model of the normal system behavior. While in the detection phase, we compare the actual system behavior with ones in the normal system. If the deviation is huge enough an alert of intrusion is sent to the network administrator or another action could be taken in order to block the intrusion. This approach saw the light thanks to Anderson [3] and Denning [4] and then applied in some IDS like IDES [5] and EMERALD [6]. The main advantage of anomaly detection is the ability to detect novel attacks. Nevertheless, this approach has some weakness, it's detection rate is modest and it produces a high false alarm rate. The main cause of this phenomenon is the manipulation of large network traffic data by the IDS.

To tackle with this limitation, many data dimensionality reduction techniques have been exploited to enhance the performance of an network IDS. The famous one, is Principal Component Analysis (PCA) used in [7] [8] [9]. PCA looks for a set of projection vectors that reduces dimensionality of original data and maximizes the variance of these projected data. The projection vectors form a low-dimensional linear subspace that allows us to effectively get the data structure from the original space.

Unfortunately, the classical PCA which is based on the Euclidean norm (L2-norm), is sensitive to data points that deviates widely from the rest of the data. These points, known as outliers, can affect significantly the PCA objective function due to the employ of large L2-norm. As a consequence, we can get inaccurate detection results. To handle this problem, several robust PCA variants have been developed [10] [11].

In this paper, especially in the context of intrusion detection, we propose to use L1-norm PCA [12], a particular form of Lp-norm PCA [13], which maximizes L1-norm based dispersion in the feature space. The solution of the proposed L1-norm PCA optimization problem is implemented using a gradient ascent method.

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 will present the L1-norm PCA as a particular case of the Lp-norm based optimization problem in Section III. The Section IV is dedicated to present the KDDcup99 dataset for intrusion detection on which we have conducted our experiments. In Section V, we will expose the experimental results, demonstrate the effectiveness of L1 norm PCA and illustrate its superiority compared to the conventional PCA. Finally, section VI summaries the principal obtained results and concludes the paper with a discussion about future research directions.

978-1-4673-8715-6/15/$31.00 ©2015 IEEE

II. Mathematical formulation of the problem

Suppose we have $X = [x_1, \ldots, x_N] \in \mathbb{R}^{d \times N}$ a data matrix, where d and N represent the dimension and total number of the samples x_i respectively. Recall that, the columns of X contain the N data samples, we subtract the average from each entry to ensure zero mean across the columns.

The classical PCA tries to find $m < d$ orthonormal projection vectors $W = [w_1, \ldots, w_m] \in \mathbb{R}^{d \times m}$, $W^T W = I_m$, such that the objective function (1) is maximized.

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

Where, I_m is the $m \times m$ identity matrix, $C = XX^T$ represents the covariance matrix of X. The operator $||.||_2$ is the Euclidean norm of a vector also known as L2-norm and tr(.) is the trace operator.

The global optimal solution of (1) is reached by solving the eigenvalue problem $CW = W\Lambda$. The solution can be reached by tacking the top eigenvectors corresponding to the largest eigenvalues contained in Λ. To do this, we can perform a singular value decomposition on the matrix C and the principal components (PC) will be the columns of the orthogonal matrix, W. Finally, we can project the original data onto the directions described by the principal components as follows $Y = W^T X$.

We can note that, samples with large norms dominate the total scatter S in (1). As a result, the process of finding the orthonormal projection vectors will not be accurate. In order to solve this, we propose to replace the L2-norm by the L1-norm which is more insensitive to outliers [13].

III. The proposed solution

Motivated by above mentioned issues, we can use the Lp-norm instead of L2-norm in the objective function (1) and then we can get the following alternative optimization problem.

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

Where, $W = [w_1, \ldots, w_m] \in \mathbb{R}^{d \times m}$ is the projection matrix whose columns w_i varies with different m's. Finding a global solution of (2) for $m > 1$ is very difficult. To remediate this problem, we divide the optimization process into two steps. Firstly, we search the optimal solution in the case of $m = 1$ and obtain only one vector w. Secondly, this solution can be extended to pick up a m vectors by applying the same above procedure greedily.

In the following subsection an algorithm to solve (2) for $m = 1$ and a greedy search algorithm for $m > 1$ are presented.

A. Solution for m=1

To simplify the problem in (2), we will search just one projection vector (m=1) then (2) becomes:

$$w^* = \arg\max_w F_p(w) = \arg\max_w \frac{1}{p} \sum_{i=1}^{N} |w^T x_i|^p \qquad (3)$$

We deal with this optimization problem by taking the gradient of $F_p(w)$ with respect to w. Nevertheless, the gradient may not be well defined with the absolute value operation. That's why we introduce the sign function as follows:

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

the sign function allow us to rewrite $F_p(w)$ in (3) as:

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

Computing the gradient of $F_p(w)$ with respect to w, we obtain

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

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

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

A special case takes place when $p \leq 1$. Here, the gradient is not well defined for $w's$ if there exist some $x_i's$ such that $w^T x_i = 0$. Nonetheless, this singular case can be skipped by slightly moving w using a small random vector δ, and this manipulation is allowed because the number of samples is limited. The steepest gradient method is used to get the projection that maximizes the objective function (3). Then, the entire optimization procedure is presented in the Algorithm 1.

Algorithm 1 : L1-norm PCA algorithm

1) Initialization: $t \leftarrow 0$. Set $w(0)$ such that $||w(0)||_2 = 1$
2) Check singularity case (if $p \leq 1$):
   - If $\exists i$, such that $w^T(t) x_i = 0$, $w(t) \leftarrow (w(t) + \delta)/||w(t) + \delta||_2$. Here, δ is a small random vector.
3) Computation of gradient ∇_w using (7).
4) $w(t+1) \leftarrow w(t) + \alpha \nabla_w$.
5) Normalization:
   - $t \leftarrow t + 1$
   - $w(t) \leftarrow w(t)/||w(t)||_2$
6) Convergence check:
   - If $||w(t) - w(t-1)||_2 > \epsilon$ goto Step 2.
   - Else, $w^* \leftarrow w(t)$. Stop iteration.

Moreover, we note that if the learning rate α is high, the steepest gradient method finds a big difficulty to converge, whereas with small values of α the convergence is not fast enough. In this paper, the parameter α is set to $\frac{10}{N}$, where N is the number of training samples.

B. Solution for $m > 1$

In this subsection, the previous L1-norm PCA algorithm is generalized to get multiple projection vectors w_i with $(m > 1)$. In summary, we present the procedure for obtaining W in Algorithm 2.

Algorithm 2 : L1-norm PCA algorithm for $m > 1$

1) Let $w_0 = 0$ and $X_0 = X$.
2) For i = 1..m
   - Set $X_i = (I_d - w_{i-1}w_{i-1}^T)X_{i-1}$
   - Apply L1-norm PCA on X_i.
3) Output $W^* = [w_1, ..., w_m]$

IV. KDDcup99 database

The KDDcup99 dataset [14] is the most popular database that has ever been used in the intrusion detection field.

The entire training dataset contained about 5,000,000 connection records. In this paper, we work only with the 10% training dataset consisted of 494,021 records which contain 97,278 normal connections (i.e. 19.69%). Each TCP connection record is composed of 41 different attributes that describe 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 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 is composed of 311,029 connections. It is important to note that the test data includes some specific attack types which doesn't exist in the training data.

Moreover, since the KDDcup99 dataset is composed of continuous and discrete attributes values, we have transformed the discrete attributes values to continuous values by using the same transformation concept used in [7].

V. Experiments and discussion

In this section, we perform a set of experiments to evaluate the performance of the proposed IDS model when applying L1-norm PCA in combination with K-NN classifier. 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 use the following measures to evaluate the performance of IDS:

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

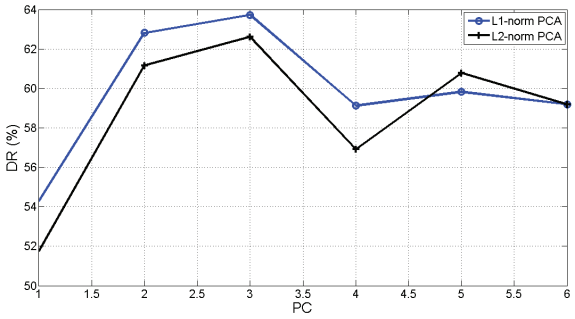
$$FPR = \frac{FP}{FP + TN} * 100 \tag{9}$$

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 DR and a low FPR.

As a first experience 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% 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.

After applying L1-norm PCA and L2-norm 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 and Fig. 2 summarize the results of this first experience.

Fig. 1. The number of principal components vs. detection rate (%)



We can see 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 at 64% for L1-norm PCA. In the other side, the FPR is minimized to approximately 1.5%. Hence, we can conclude that the proposed L1-norm PCA is much better than classical PCA.

In order to go deeper in our investigation, we have calculated the detection rate of L1-norm PCA and L2-norm PCA for every type of attack from the four categories DOS, U2R, R2L and PROBE.

According to Table I, it is shown that the IDS which uses L1-norm PCA detects more efficiently DOS (93.65%) and

Fig. 2. The number of principal components vs. FPR

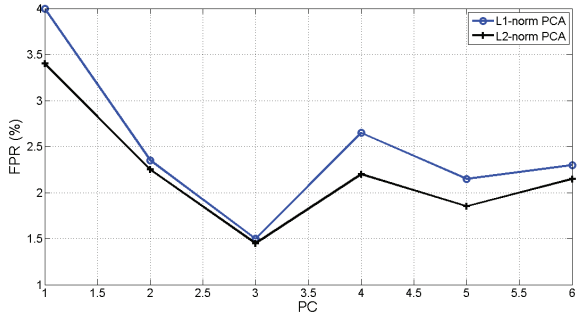


TABLE I
COMPARAISON OF ATTACKS DETECTION RATE (%) FOR L1-NORM AND L2-NORM PCA

PCA	Normal	DOS	U2R	R2L	PROBE
L1-norm	98.5	**93.65**	11.2	**4.2**	**75.5**
L2-norm	98.55	92.85	11	4.2	65.65

PROBE attacks (75.5%), even if the U2R and R2L attacks are similarly detected for both variants of PCA.

In the next experiment, we have varied the size of the training dataset to see its effect on the DR and FPR of our IDS. As illustrated in Fig. 3 and Fig. 4, when we increase

Fig. 3. Training data vs. Detection rate

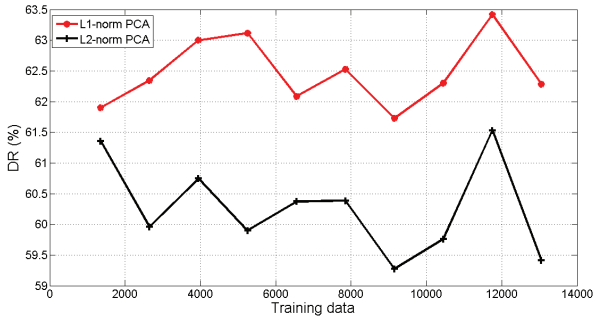
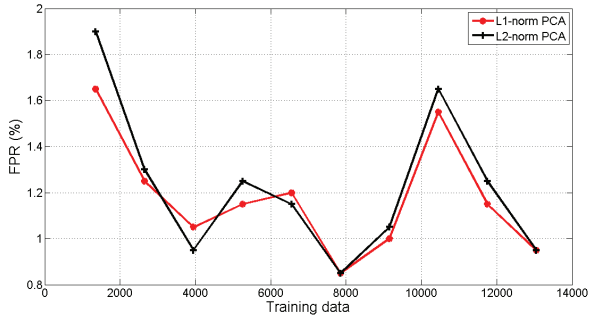


Fig. 4. Training data vs. FPR



the training data size, the detection rate increases and reach its maximum at 63.5% for L1-norm PCA but only 61.5% for PCA.

In the same manner, we have calculated the detection rate by varying the number of connection records in the training dataset to see its effect on the performance of the IDS.

According to the Table II, we note that DOS and PROBE attacks are well detected by the IDS especially with L1-norm PCA. However, the U2R and R2L attacks are not suitably detected because the KDDcup99 dataset contains a few connection records for these type of attack. The obtained results confirms that the proposed system has good performance from the point of view of a better attacks detection and lower false alarms.

TABLE II
TRAINING DATA SIZE VS. ATTACK'S DETECTION RATE (%)

Training data records	PCA	DOS	U2R	R2L	PROBE
2650	L1-norm	**91.05**	**12.9**	**4.35**	**76.8**
	L2-norm	89.05	12.60	4.30	64
3950	L1-norm	**93.7**	**12.10**	**4.2**	**77.15**
	L2-norm	91.20	11.8	4.15	64.90
5250	L1-norm	**92.20**	**10.80**	**4.45**	**80**
	L2-norm	89.20	10.80	4.45	65.35
6550	L1-norm	**92.95**	**9**	**3.8**	**80.3**
	L2-norm	90.20	9	3.75	70.45
7850	L1-norm	**92.45**	**10.90**	**5.45**	**78.85**
	L2-norm	90.15	10.30	5.45	66.95
9150	L1-norm	**91.95**	**9.9**	**5**	**79.55**
	L2-norm	89.15	9.6	4.95	65.70
10450	L1-norm	**92.65**	**9.8**	**5.3**	**78.80**
	L2-norm	91.10	9.50	5.20	65.40
11750	L1-norm	**94.30**	**9.80**	**4.80**	**79.80**
	L2-norm	92.25	9.2	4.75	67.05
13050	L1-norm	**92.45**	**9.70**	**4.80**	**80**
	L2-norm	90.90	9.20	4.80	64.60

Unfortunately, we have noted that the CPU time consumed by L1-norm PCA can be seen as its major weakness. The main reason behind this drawback is that the projection vectors w_i are extracted one by one greedily.

VI. CONCLUSION

Classical Principal component analysis (PCA) is sensitive to outliers, defined as the samples that deviate significantly from the rest of the data. As a solution to handle this weakness, we have proposed L1-norm PCA, a technique which tries to find projections that maximize total covariance using L1-norm instead of L2-norm. Experimental results showed that, L1-norm PCA takes advantage over PCA in detection of all type of attacks. Furthermore, PCA-L1 gives low false positive alarms. However, this approach consumes more CPU time compared to classical PCA due to the greediness nature of the algorithm. Our future works will be oriented towards other optimization techniques, which are faster than the steepest gradient and make the work of L1-norm PCA much faster and accurate.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the manuscript.

References

[1] S. Kumar and E. Spafford, "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. Anderson, "Computer security threat monitoring and surveillance," Technical report, James P. Anderson Company, Fort Washington, Pennsylvania, Tech. Rep., 1980.

[4] D. E. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering,*, no. 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). sri international," Computer Science Laboratory, SRI International, Menlo Park, California,, Tech. Rep., 1992.

[6] P. A. Porras and P. G. Neumann, "Emerald: Event monitoring enabling response to anomalous live disturbances," in *Proceedings of the 20th national information systems security conference*, 1997, pp. 353–365.

[7] Y. Bouzida, F. Cuppens, N. Cuppens-Boulahia, and S. Gombault, "Efficient intrusion detection using principal component analysis," in *3éme Conférence sur la Sécurité et Architectures Réseaux (SAR), La Londe, France*, 2004.

[8] 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.

[9] W. Wang and R. Battiti, "Identifying intrusions in computer networks with principal component analysis," in *Proceedings of The First International Conference on Availability, Reliability and Security, ARES* . IEEE, 2006, pp. 8–pp.

[10] H. Xu, C. Caramanis, and S. Mannor, "Outlier-robust pca: The high-dimensional case," *IEEE Transactions on Information Theory,*, vol. 59, no. 1, pp. 546–572, 2013.

[11] C. Pascoal, M. Oliveira, A. Pacheco, and R. Valadas, "Detection of outliers using robust principal component analysis: A simulation study," in *Combining Soft Computing and Statistical Methods in Data Analysis*. Springer, 2010, pp. 499–507.

[12] N. Kwak, "Principal component analysis based on l_1-norm maximization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 9, pp. 1672–1680, 2008.

[13] ——, "Principal component analysis by l_p -norm maximization," *IEEE Transactions on, Cybernetics*, vol. 44, no. 5, pp. 594–609, May 2014.

[14] [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/