A Fast Algorithm for Detecting Anomalous Changes in Network Traffic

Tingshan Huang, Harish Sethu and Nagarajan Kandasamy

ECE Department, Drexel University
Philadelphia, PA 19104, USA
Email: { th423, sethu, kandasamy}@drexel.edu

***Abstract*—Anomalies in communication network traffic caused by malware or denial-of-service attacks manifest themselves in structural changes in the covariance matrix of traffic features. Real-time detection of anomalies in high-dimensional data demands a very efficient algorithm to identify these changes in a compact low-dimensional representation. This paper presents an efficient algorithm for the rapid detection of structural differences between two covariance matrices, as measured by the maximum possible angle between the subspaces specified by subsets of the two sets of principal components of the matrices. We show that our algorithm achieves a significantly lower computational complexity compared to a naive approach. Finally, we apply our results to real traffic traces from Internet backbone links and show that our approach offers a substantial reduction in the computational overhead of anomaly detection.**

I. Introduction

Security professionals who monitor communication networks for malware or denial-of-service attacks are increasingly dependent on real-time detection of anomalous behavior in the traffic data. Such detection and classification of threats even as they are unfolding requires a fast and efficient method to assess changes in traffic features over very short intervals of time.

This work is primarily motivated by two observations: (i) anomalies lead to changes in the covariance matrix of the set of traffic features being monitored, and (ii) different types of anomalies cause different types of deviations in the covariance matrix allowing a categorization of the detected anomaly and an immediate prescription of actions toward threat mitigation [1]. This is to be expected since the covariance matrix, as a second-order statistic, captures the variance of each monitored feature as well as the correlation between each pair of features. Besides, use of the covariance matrix requires no assumptions about the distributions of the monitored features, making it a general method for traffic characterization. For these reasons, this paper is focused on characterizing communication traffic as a stream of covariance matrices, one for each designated window of time, and then using observed changes in the covariance matrices to infer changes in the system status.

Current state-of-the-art network traffic analysis invariably deals with high-dimensional datasets of increasingly larger size—thus, it is important to derive a low-dimensional structure as a compact representation of the original dataset. Principal component analysis (PCA) allows us to examine the linear relationship between features of traffic and derive a reduced set of unrelated features that are linear combinations of the original features [2]. In this work, we detect and quantify differences between two covariance matrices by the changes in the principal components of the covariance matrices.

This paper introduces a new metric for detecting changes based on the angle between the subspaces specified by the most significant principal components of two given matrices. We also present an efficient algorithm for computing this metric. The output of this algorithm also includes two sets of distinguishing characteristics for the two matrices which can be employed for purposes such as anomaly detection. As a result, the algorithm supports the real-time detection of anomalies by allowing rapid detection and categorization of the structural differences between the covariance matrices.

II. Problem statement

Let N denote the number of features in the data set of interest. Let Σ_A and Σ_B denote the two $N \times N$ covariance matrices to be compared. These two matrices could represent real traffic data during different time windows or one of them could be a reference matrix representing normal operation without anomalies. Let $\mathbf{a}_1, \ldots, \mathbf{a}_N$ and $\mathbf{b}_1, \ldots, \mathbf{b}_N$ denote the eigenvectors of Σ_A and Σ_B, respectively.

Let $\theta_{k_A,k_B}(A,B)$ denote the angle between the subspace composed of the first k_A principal components of Σ_A, $\mathbf{a}_1, \ldots, \mathbf{a}_{k_A}$, and the subspace composed of the first k_B principal components of Σ_B, $\mathbf{b}_1, \ldots, \mathbf{b}_{k_B}$. We refer to this angle between the subspaces as the *subspace distance*, which has a range between 0 to 90 degrees. We have:

$$\sin\theta_{k_A,k_B}(A,B) = \parallel T_{k_A,k_B}(A,B)) \parallel \tag{1}$$

where $\parallel \cdot \parallel$ is the matrix norm and $T_{k_A,k_B}(A,B)$ is the part of $[\mathbf{b}_1, \ldots, \mathbf{b}_{k_B}]$ orthogonal to $[\mathbf{a}_1, \ldots, \mathbf{a}_{k_A}]$. Therefore,

$$T_{k_A,k_B}(A,B) = (I - \sum_{i=1}^{k_A} \mathbf{a}_i \times \mathbf{a}_i')[\mathbf{b}_1, \ldots, \mathbf{b}_{k_B}] \tag{2}$$

$$= (\sum_{i=k_A+1}^{N} \mathbf{a}_i \times \mathbf{a}_i')[\mathbf{b}_1, \ldots, \mathbf{b}_{k_B}]. \tag{3}$$

To compare the two matrices, we quantify the difference between Σ_A and Σ_B as the maximum angle $\theta_{k_A,k_B}(A,B)$, where $1 \leq k_A \leq N$ and $1 \leq k_B \leq N$:

$$\theta_{max} = \max_{1 \leq k_A, k_B \leq N} \theta_{k_A,k_B}(A,B) \tag{4}$$

978-3-901882-77-7

In this paper, we refer to θ_{max} as the *maximum subspace distance*, which quantifies the difference between the two matrices.

When k_A and k_B hold values that maximize $\theta_{k_A,k_B}(A,B)$, the two sets of principal components, $[\mathbf{a}_1,\ldots,\mathbf{a}_{k_A}]$ and $[\mathbf{b}_1,\ldots,\mathbf{b}_{k_B}]$, can be thought of as the distinguishing characteristics of covariance matrices Σ_A and Σ_B. We show in Section IV how to employ these two sets of characteristics for change detection.

The problem considered in this paper is one of estimating k_A and k_B which maximize the subspace distance without the overhead of computing $\theta_{k_A,k_B}(A,B)$ for all k_A and k_B.

III. Solution

Our solution is based on four key ideas: (i) allowing $k_A = k_B$ in our search for the maximum subspace distance, θ_{max}, (ii) reducing the problem to one of always finding only the first principal component of a matrix, (iii) using the power iteration method to approximate the first principal components, and finally, (iv) using a heuristic to approximately ascertain the value of θ_{max}.

A. The rationale behind allowing $k_A = k_B$

In the approach presented in this paper, we limit our search for θ_{max} to consider only the cases in which $k_A = k_B$. Our rationale is based on Theorem 1 below.

Theorem 1. If for some k_A and k_B, $\theta_{k_A,k_B}(A,B) = \theta_{\max}$, then there exists k, $1 \le k \le N$, such that $\theta_{k,k}(A,B) = \theta_{\max}$.

Proof: Without loss of generality, assume $k_A \ge k_B$. The statement of the theorem is proved if $\theta_{k_A,k_B}(A,B) \le \theta_{k_B,k_B}(A,B)$ and $\theta_{k_A,k_B}(A,B) \le \theta_{k_A,k_A}(A,B)$. The proofs of each of these two cases follows.

Case (i): Let $\mathbf{x}$ denote a column vector of length k_B. Using the definition of matrix norm, we have:

$$\begin{aligned}
&\| T_{k_A,k_B}(A,B) \|^2 \\
&= \max_{\forall \mathbf{x}, \|\mathbf{x}\|=1} \| T_{k_A,k_B}(A,B)\mathbf{x} \|^2 \\
&= \max_{\forall \mathbf{x}, \|\mathbf{x}\|=1} \sum_{i=k_A+1}^{N} \| \mathbf{a}_i'[\mathbf{b}_1,\ldots,\mathbf{b}_{k_B}]\mathbf{x} \|^2 \\
&= \sum_{i=k_A+1}^{N} \| \mathbf{a}_i'[\mathbf{b}_1,\ldots,\mathbf{b}_{k_B}]\mathbf{x}_{max} \|^2
\end{aligned}$$

where $\mathbf{x}_{max}$ is the column vector with unit norm that achieves the maximum matrix norm. Similarly, using the definition of the matrix norm again:

$$\begin{aligned}
&\| T_{k_B,k_B}(A,B) \|^2 \\
&= \max_{\forall \mathbf{x}, \|\mathbf{x}\|=1} \sum_{i=k_B+1}^{N} \| \mathbf{a}_i'[\mathbf{b}_1,\ldots,\mathbf{b}_{k_B}]\mathbf{x} \|^2 \\
&\ge \sum_{i=k_B+1}^{N} \| \mathbf{a}_i'[\mathbf{b}_1,\ldots,\mathbf{b}_{k_B}]\mathbf{x}_{max} \|^2 \\
&\ge \sum_{i=k_A+1}^{N} \| \mathbf{a}_i'[\mathbf{b}_1,\ldots,\mathbf{b}_{k_B}]\mathbf{x}_{max} \|^2 = \| T_{k_A,k_B}(A,B) \|^2 .
\end{aligned}$$

Thus we have $\theta_{k_A,k_B}(A,B) \le \theta_{k_B,k_B}(A,B)$.

Case (ii): Let $\mathbf{y}$ denote a column vector of length k_A. Using the definition of matrix norm:

$$\| T_{k_A,k_A}(A,B) \|^2 = \max_{\forall \mathbf{y}, \|\mathbf{y}\|=1} \sum_{i=k_A+1}^{N} \| \mathbf{a}_i'[\mathbf{b}_1,\ldots,\mathbf{b}_{k_A}]\mathbf{y} \|^2$$

Let $\mathbf{z}$ denote a column vector of length k_A with $\mathbf{x}_{max}$ followed by $k_A - k_B$ zeroes: $\mathbf{z}' = [\mathbf{x}_{max}\ 0\cdots 0]'$. This vector has unit norm and $[\mathbf{b}_1,\ldots,\mathbf{b}_{k_A}]\mathbf{z} = [\mathbf{b}_1,\ldots,\mathbf{b}_{k_B}]\mathbf{x}_{max}$. Therefore,

$$\begin{aligned}
\| T_{k_A,k_A}(A,B) \|^2 &\ge \sum_{i=k_A+1}^{N} \| \mathbf{a}_i'[\mathbf{b}_1,\ldots,\mathbf{b}_{k_A}]\mathbf{z} \|^2 \\
&= \sum_{i=k_A+1}^{N} \| \mathbf{a}_i'[\mathbf{b}_1,\ldots,\mathbf{b}_{k_B}]\mathbf{x}_{max} \|^2 = \| T_{k_A,k_B}(A,B) \|^2 .
\end{aligned}$$

Thus we have $\theta_{k_A,k_B}(A,B) \le \theta_{k_A,k_A}(A,B)$. ■

Allowing $k_A = k_B$ to find the maximum subspace distance reduces the search space from N^2 to N. We refer to the value of k for which $\theta_{k,k}(A,B)$ is the maximum subspace distance as the *optimal subspace dimension*.

B. Estimating the optimal subspace dimension

It is straightforward to find the optimal subspace dimension by computing $\theta_{k,k}(A,B)$ for every k from 1 to N and determining the k for which $\theta_{k,k}(A,B)$ is the maximum. However, we are more interested in the choice of a smaller k, which we will call the *effective subspace dimension* and which allows a similar ability to distinguish between the two subspaces as the optimal subspace dimension does.

Our algorithm computes an approximate value of $\theta_{k,k}(A,B)$ beginning with $k = 1$ and stops when the subspace distance, $\theta_{k,k}(A,B)$, has dropped for the d-th time where d is an input to the algorithm. In our experiments on Internet backbone traffic traces, discussed in Section IV, we use $d = 3$. The maximum observed $\theta_{k,k}(A,B)$ is used as the *estimated* maximum subspace distance and the corresponding k becomes the effective subspace dimension.

Fig. 1 presents the pseudo-code of our algorithm, *getESD*, which returns the effective subspace dimension and the estimated maximum subspace distance. During the k-th iteration of the *while* loop, we compute the k-th eigenvector for Σ_A and Σ_B ($\mathbf{a}_k$ and $\mathbf{b}_k$), and use the two sets of k eigenvectors ($[\mathbf{a}_1,\ldots,\mathbf{a}_k]$ and $[\mathbf{b}_1,\ldots,\mathbf{b}_k]$) to calculate the new subspace distance (θ') between them. Note that the k-th eigenvectors for Σ_A and Σ_B are not computed until we are in the k-th iteration of the *while* loop. Our implementation of this algorithm uses the power iteration method [3] for computing the eigenvectors and the norm of T_k.

C. Complexity analysis

The computational complexity of *getESD* is composed of (i) the computation of T_k in k loops which is $\mathcal{O}(k^3N^2)$, where k is the effective subspace dimension (ii) the computation of the first k eigenvectors of Σ_A and Σ_B using the power iteration method, which is $\mathcal{O}(kPN^2)$ where P is a tuning parameter,

procedure GETESD(Σ_A, Σ_B, d)
  $k \leftarrow 1$ ▷ number of Principal Components (PCs)
  $\theta \leftarrow 0$ ▷ angle between two subspaces
  $\hat{\Sigma}_A \leftarrow \Sigma_A$ ▷ projection of Σ_A on its last $N-k$ PCs
  $\hat{\Sigma}_B \leftarrow \Sigma_B$ ▷ projection of Σ_B on its last $N-k$ PCs
  $\theta_{\max} \leftarrow 0$ ▷ maximum angle observed
  ESD $\leftarrow 0$ ▷ value of k corresponding to $\theta_{\max}$
  numDrops $\leftarrow 0$ ▷ number of decreases in θ
  while $(k \leq N)$ **do**
    $\mathbf{a}_k \leftarrow$ estimated first PC of $\hat{\Sigma}_A$ ▷ k-th PC of Σ_A
    $\mathbf{b}_k \leftarrow$ estimated first PC of $\hat{\Sigma}_B$ ▷ k-th PC of Σ_B
    $T_k \leftarrow [\mathbf{b}_1, \ldots, \mathbf{b}_k] - \sum_i^k \mathbf{a}_i \times (\mathbf{a}_i' \times [\mathbf{b}_1, \ldots, \mathbf{b}_k])$
    norm(T_k) $\leftarrow$ sqrt(estimated first eigenvalue of $T_k T_k'$)
    $\theta' \leftarrow \arcsin(\text{norm}(T_k))$
    if $(\theta' < \theta)$ **then**
      numDrops $\leftarrow$ *numDrops* $+1$
      if $(\theta > \theta_{\max})$ **then**
        $\theta_{\max} \leftarrow \theta$
        ESD $\leftarrow k-1$
      end if
      if (*numDrops* $== d$) **then**
        return (*ESD*, $\theta_{\max}$)
      end if
    end if
    $\hat{\Sigma}_A \leftarrow \hat{\Sigma}_A - \mathbf{a}_k \times (\mathbf{a}_k' \times \hat{\Sigma}_A)$
    $\hat{\Sigma}_B \leftarrow \Sigma_B - \mathbf{b}_k \times (\mathbf{b}_k' \times \hat{\Sigma}_B)$
    $k \leftarrow k+1$
    $\theta \leftarrow \theta'$
  end while
  return $(N-1, \theta_{\max})$
end procedure

Fig. 1. The *getESD* algorithm.

and (iii) the computation of $\| T_k \|$ in k loops, which is $\mathcal{O}(kPN^2)$.

Since $P \ll N$, the computation of T_k becomes the dominant complexity. The computational complexity of the *getESD* algorithm, therefore, is $\mathcal{O}(k^3N^2)$.

For the naive method which computes all eigenvectors, the complexity is again dominated by the computation of T_k in N loops, which is $\mathcal{O}(N^5)$. Since the effective subspace dimension, k, is much smaller than N, the computational complexity, $\mathcal{O}(k^3N^2)$, of the *getESD* algorithm is a significant improvement.

IV. Results with Internet Backbone Traffic

We use anonymized passive traffic traces from four monitors of CAIDA (Cooperative Association for Internet Data Analysis) connected to high-speed Internet backbone links of Tier1 ISPs: equinix-chicago-dirA, equinix-chicago-dirB, equinix-sanjose-dirA, and equinix-sanjose-dirB [4]. The following six features of each packet are selected for data analysis: source IP address, destination IP address, source port number, destination port number, protocol and packet size. This allows a total of $4 \times 6 = 24$ features. We compute the 24×24 covariance matrix of these features for every time window of 50 milliseconds and compare it to the covariance matrix corresponding to the previous time window.

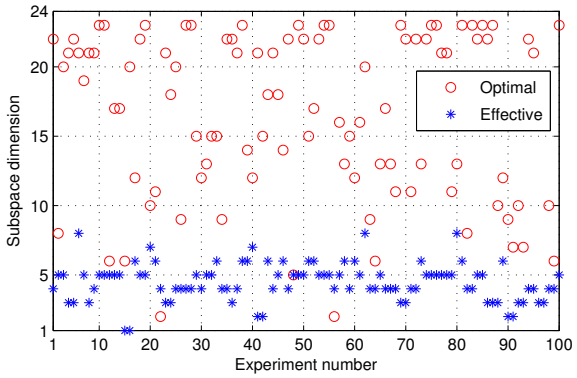


Fig. 2. The optimal and the effective subspace dimension for estimating the maximum subspace distance between covariance matrices corresponding to consecutive time windows in Internet backbone traffic traces.

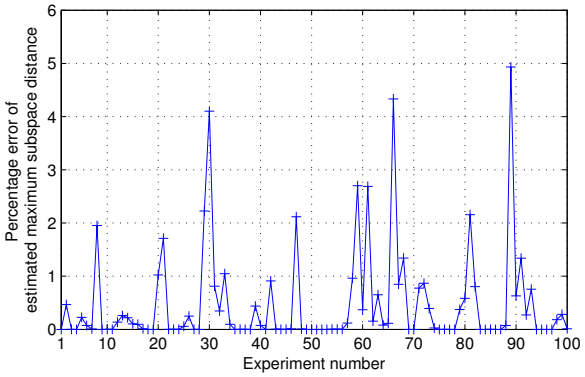


Fig. 3. Relative percentage error of the estimated maximum subspace distance between covariance matrices corresponding to consecutive time windows in Internet backbone traffic traces.

A. Estimation of subspace distance

Our results with Internet backbone traffic data show that the maximum subspace distance between two covariance matrices corresponding to consecutive time windows has a mean of 89.82 degrees and is close to 90 degrees with high probability. This shows that we can always choose two sets of principal components—one set for each covariance matrix with size equal to the optimal subspace dimension—that are almost orthogonal to each other in order to capture and successfully detect anomalous changes.

Fig. 2 shows that the effective subspace dimension rarely exceeds 8 with an average of 4.33, a significant reduction compared to the 24 principal components that would have to be examined to find the actual maximum subspace distance.

As also shown in Fig. 2, the effective subspace dimension is almost always smaller than the optimal subspace dimension, which has an average of about 16.5 and can be as high as 23. Even though it is true in the case of most real data that the first few principal components are the most significant in capturing the internal structure of the data while the last few are comparatively trivial, Fig. 2 shows that the optimal subspace dimension cannot often be computed by considering only a few fixed number of significant principal components. The *getESD* algorithm, therefore, offers a solution to determine

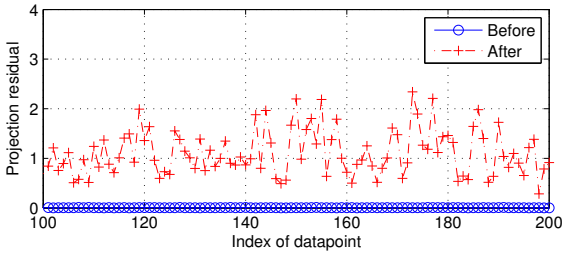
an effective number of principal components to obtain a good approximation of the maximum subspace distance.

Fig. 3 shows that the relative percentage error of the estimated maximum subspace distance computed by the *getESD* algorithm rarely exceeds 5%. In fact, the average relative percentage error is as small as 0.64%.

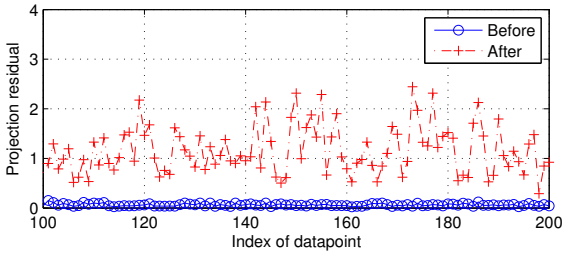
B. Anomaly detection using Projection Residual

To demonstrate the effectiveness of anomaly detection with the *getESD* algorithm, we use two synthetic datasets labeled *Before* and *After*, drawn from a random Gaussian distribution with zero mean, but with different covariance matrices. These covariance matrices are such that 10% of the principal components capture more than 99.9% of the total variance. We use the effective subspace dimension to construct a normal subspace that consists of the first few principal components of the covariance matrix of dataset *Before*, assumed to be normal. We then project the dataset *Before* and dataset *After* onto the normal subspace, and use the projection residual to indicate the level of abnormality. The projection residual of the datasets *Before* and *After* using the *getESD* algorithm are shown in Fig. 4(a). It is clear that the dataset *Before* has small projection residuals, which is expected; the dataset *After* has large projection residuals, indicating a difference in its properties with regard to the covariance matrix.

Fig. 4(b) presents results with the subspace method [5], where the dimension of normal subspace captures 99.5% of the variance in the dataset. The projection residual result of subspace method is similar to that of our *getESD* algorithm.



(a) Using the effective subspace dimension obtained from the *getESD* algorithm.



(b) Using the minimum number of principal components that captures 99.5% of the variance in the dataset.

Fig. 4. Projection residual of datasets BEFORE and AFTER onto normal subspace.

V. Related Work

The covariance matrices can be directly employed to detect matrices without applying PCA. For example, [1] proposes the detection of network anomalies by monitoring changes in covariance matrix, where test covariance matrices are compared against covariance matrices under normal conditions. Other methods, such as in [6], also avoid using PCA directly and instead employ graph-based filters to project the graph signals on normal and anomaly subspaces to inform the detection of anomalies. Our work, on the other hand, exploits the correlation of the changes amongst features as indicated by the changes in the key principal components.

The work presented in this paper is most closely related to the PCA-subspace approach for network-wide anomaly detection first proposed by Lakhina *et al.* [5], [7]. The approach, based on detecting deviations in the traffic volume and feature distributions caused by anomalies, does have its limitations [8], [9]. For example, when a large anomaly dominates in a certain dimension and disguises itself as normal traffic, it may succeed in foiling the PCA-based approach [8]. Consequently, there have been many recent extensions of the PCA-subspace approach [10]–[15].

An underlying assumption of the original PCA-subspace method introduced by Lakhina is that the dominant principal components define normal behavior in network traffic. It is argued in [9] that the characteristics of the origin-destination flows of a network determine if a small number of eigenflows can capture most of the variance. As a result, many anomalies which result in a large number of small flows will be missed using this method. Our work presented in this paper, on the contrary, imposes no such assumptions on the normal behavior of network traffic.

As stated in [8], the performance of the PCA-based method is sensitive to the dimensions chosen for the normal subspace. One may use a specified number of dimensions, as in [10], or choose the number of dimensions that accounts for a specified percentile of the variation, as in [12] and [14]. Other methods exploit the most relevant principal components, as in [16], while yet others retain the components with the lowest variance as the extracted features [17]. This paper complements these recent approaches because, when comparing the covariance matrices under normal and abnormal conditions, the effective subspace dimension returned by our algorithm can be interpreted as the right number of dimensions for both the normal and abnormal behavior of network traffic. Instead of defining the characteristics of normal behavior only, as in [5] and [7], our work points out the characteristics of both normal and abnormal behaviors through two sets of principal components.

VI. Concluding Remarks

The *getESD* algorithm presented in this paper reduces the computational overhead of comparing two covariance matrices of traffic features for real-time anomaly detection. Meanwhile, it also retains an ability to distinguish between matrices that is close to what an optimal algorithm would accomplish. The algorithm is a general-purpose tool for comparing structural differences between evolving matrices and is applicable in almost any scenario that involves processing a stream of matrices for real-time detection of developing changes.

Acknowledgements: This work was partially supported by the National Science Foundation Award #1228847.

REFERENCES

[1] D. Yeung, S. Jin, and X. Wang, "Covariance-matrix modeling and detecting various flooding attacks," *IEEE Transactions on Systems, Man and Cybernetics, Part A*, vol. 37, no. 2, pp. 157–169, Feb. 2007.

[2] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. New York: Springer, 2002.

[3] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Johns Hopkins Univ. Press, 1989.

[4] CAIDA, "The CAIDA Anonymized Internet Traces 2013." [Online]. Available: http://www.caida.org/data/passive/passive_2013_dataset.xml

[5] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, pp. 219–230, Aug. 2004.

[6] H. Egilmez and A. Ortega, "Spectral anomaly detection using graph-based filtering for wireless sensor networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 1085–1089.

[7] A. Lakhina, M. Crovella, and C. Diot, "Mining anomalies using traffic feature distributions," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 4, pp. 217–228, Aug. 2005.

[8] H. Ringberg, A. Soule, J. Rexford, and C. Diot, "Sensitivity of PCA for traffic anomaly detection," *ACM SIGMETRICS Performance Evaluation Review*, vol. 35, no. 1, pp. 109–120, June 2007.

[9] B. Zhang, J. Yang, J. Wu, D. Qin, and L. Gao, "PCA-subspace method—is it good enough for network-wide anomaly detection," *IEEE Network Operations and Management Symposium*, pp. 359–367, Apr. 2012.

[10] T. Kudo, T. Morita, T. Matsuda, and T. Takine, "PCA-based robust anomaly detection using periodic traffic behavior," *Proc. IEEE Int'l Conf. on Communications*, pp. 1330–1334, June 2013.

[11] C. Pascoal, M. Rosario de Oliveira, R. Valadas, P. Filzmoser, P. Salvador, and A. Pacheco, "Robust feature selection and robust PCA for Internet traffic anomaly detection," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 1755–1763.

[12] S. Novakov, C.-H. Lung, I. Lambadaris, and N. Seddigh, "Studies in applying PCA and wavelet algorithms for network traffic anomaly detection," *IEEE Int'l. Conf. on High Performance Switching and Routing*, pp. 185–190, July 2013.

[13] R. Jiang, H. Fei, and J. Huan, "A family of joint sparse PCA algorithms for anomaly localization in network data streams," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 11, pp. 2421–2433, Nov. 2013.

[14] C. Callegari, L. Gazzarrini, S. Giordano, M. Pagano, and T. Pepe, "A novel PCA-based network anomaly detection," in *Proc. IEEE Int'l Conf. on Communications*, June 2011, pp. 1–5.

[15] ——, "Improving PCA-based anomaly detection by using multiple time scale analysis and kullback–leibler divergence," *International Journal of Communication Systems*, vol. 27, no. 10, pp. 1731–1751, 2014.

[16] Q. Guan and S. Fu, "Adaptive anomaly identification by exploring metric subspace in cloud computing infrastructures," in *IEEE International Symposium on Reliable Distributed Systems*, Sept 2013, pp. 205–214.

[17] L. Kuncheva and W. Faithfull, "PCA feature extraction for change detection in multidimensional unlabeled data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 1, pp. 69–80, Jan 2014.