# Technical Report: Networkmetrics. Multivariate Visual Analytics for Networking Data

August 5, 2014

José Camacho
Dpt. of Signal Theory, Telematics and Communications
ETSIIT - CITIC - University of Granada (Spain)
josecamacho@ugr.es

**Abstract**

This report presents a framework for time series data analysis with application to networking data. The framework combines the powerful analysis capabilities of multivariate models based on projection subspaces with the benefits of data visualization. The approach is based on a recently proposed framework for Exploratory Data Analysis (EDA), which main goal is the knowledge generation. This is tackled using a number of appropriate techniques to maximize insight into a data set and uncover the underlying structure, including the identification of relevant features and abnormal observations. The approach is context free, so that it can be equally applied to different sources of data: traffic data, different types of logs, network management data, etc. This is especially convenient for a network manager since the number of tools needed to be handled and understood is drastically reduced. Results show that with only a few plots correctly interpreted, the user gains a detailed insight into the data, including valuable information for data modelling and decision making.

**Keywords:** Multivariate Analysis, Latent Structures, Exploratory Data Analysis, Traffic Analysis, Time Series

## 1 Introduction

Network traffic analysis is a field of constant evolution as a consequence of the evolution of the communication systems and protocols. In this changing context, traffic monitoring is an issue of great concern in order to understand and optimize the telecommunication technologies [1, 2]. The successful monitoring of networking data sets requires the parsing and analysis online of tons of time series data. There is a clear interest in developing methods to manage these scales of data while taking advantage of *the basic importance of simply looking at data* [3]. This interest has lead to the development of sophisticated visualization tools within the so-called visual analytics research area [4]. After all, *a picture is worth a thousand log entries* [5].

The present work proposes the application of Exploratory Data Analysis (EDA) based on multivariate models to the analysis of networking data. Because this report presents an extension of techniques with an important impact in research areas such as chemometrics and psychometrics to networking data, the approach is named networkmetrics.

# 2 Multivariate Techniques based on Projection Subspaces

Multivariate techniques based on projection subspaces are typically applied to two-way data sets, although extensions of these models to multi-way data sets also exist [6]. Most of the data collected in a problem domain can be parsed into a two-way data set, where a number of variables or features (columns) are measured/computed for a number of observations or objects (rows). For instance, traffic data analysis can be performed by computing, for each single flow, a number of features such as mean packet and payload size, protocol, flags, etc [7]. The result can be stored in a two-way matrix where the columns are the different features and the rows represent the flows. By applying a multivariate technique to this matrix, the network manager may find the patterns in the traffic of the network.

Both PCA and PLS are two-way methods that perform a similar solution to the same problem: data collinearity in highly multidimensional data sets. Data collinearity refers to high correlation among the data variables or features. Data understanding is subject to identifying collinearity, since the latter is found when the true, latent, dimension of the data is much lower than the actual number of features. For instance, the time-series traffic load in the several routers in the path of a Denial of Service (DoS) attack may be statistically correlated. Multivariate techniques may help to elucidate this, showing that there is a single main source of traffic data.

The approach of PCA and PLS to overcome the problems derived from collinearity is to identify a reduced number of new features, referred to as latent variables (LVs) or specifically in PCA as principal components (PCs). These LVs are obtained as a combination of the original features in the data. In standard PCA and PLS, the LVs are linear combinations of the original features, but non-linear extensions also exist [8]. For a given data set, the LVs are found by maximizing a given quadratic function, variance in the case of PCA and covariance for PLS. The operation to obtain the LVs from the original features can be geometrically interpreted as a projection operation. Thus, projection models can be understood as projection subspaces of the original feature space.

## 2.1 Unsupervised analysis: PCA

The aim of PCA is to find the subspace of maximum variance in the $M$-dimensional feature space using a $N \times M$ calibration matrix $\mathbf{X}$. The original features, commonly correlated, are linearly transformed into a lower number of uncorrelated features: the PCs. The directions of the PCs are obtained from the eigenvectors of $\mathbf{X}^T \cdot \mathbf{X}$, typically for mean centered $\mathbf{X}$. Also, $\mathbf{X}$ may be scaled so that all variables have an adequate load on the final model. Without additional assumptions, the auto-scaling operation, where all variables are centered and scaled to unit variance, is commonly used.

PCA follows the expression:

$$\mathbf{X} = \mathbf{T}_A \cdot \mathbf{P}_A^t + \mathbf{E}_A, \tag{1}$$

where $\mathbf{T}_A$ is the $N \times A$ score matrix containing the projection of the observations in the $A$ PCs sub-space, $\mathbf{P}_A$ is the $M \times A$ loading matrix containing the $A$ eigenvectors of $\mathbf{X}^T \cdot \mathbf{X}$ with highest eigenvalues associated and $\mathbf{E}_A$ is the $N \times M$ matrix of residuals. The number of PCs retained in a PCA model, $A$, is a principal choice [9] [10], which in general can be regarded as an application dependent decision [11]. When no dimension reduction is done, that is $A = Rank(\mathbf{X})$, PCA is just a rotation of the axes.

PCA is suited for applications in which there is interest in splitting variance in typically two orthogonal sub-spaces: the structure or model subspace and the residual subspace. Examples of such applications are dimensionality reduction [12] [13] [14] [15] and process monitoring [16] [17]. In particular, in the context of data mining, PCA has been typically regarded as a simple preprocessing tool for dimensionality reduction. This is particularly appropriate when $\mathbf{X}$ contains highly collinear data, so that several eigenvalues in $\mathbf{X}^T \cdot \mathbf{X}$ are low enough to be considered as noise. However, in the context of EDA, PCA is a principal technique, and the information from each subspace (residual and model subspaces) may be interpreted together.

## 2.2 Supervised analysis: PLS

PLS essentially performs an alternative solution of the linear regression problem to the least squares solution. The linear regression problem is defined by the following expression:

$$\mathbf{Y} = \mathbf{X} \cdot \mathbf{B} + \mathbf{F} \tag{2}$$

where $\mathbf{Y}$ is the $N \times K$ matrix of features that are to be estimated, $\mathbf{X}$ is the $N \times M$ matrix of features available to estimate $\mathbf{Y}$, $\mathbf{B}$ is the $M \times K$ matrix of regression coefficients and $\mathbf{F}$ is the $N \times K$ matrix of residuals. A possible way to interpret $\mathbf{B}$ is as a model of $\mathbf{Y}$, being $\mathbf{X}$ the input of the model.

Ordinary Least Squares (OLS) or simply least squares performs a solution to the linear regression problem that minimizes the quadratic error. The least squares solution for (2) is:

$$\hat{\mathbf{B}}_{LS} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{Y} \tag{3}$$

The inversion of the matrix $\mathbf{X}^T \cdot \mathbf{X}$ requires it to be non-singular. Moreover, the OLS solution is highly unstable, leading to a poor estimation performance when this matrix is ill-conditioned. The good conditioning of such a matrix is dependent on the degree of independence among features in $\mathbf{X}$. Therefore, if the features of $\mathbf{X}$ are highly collinear, the estimation of $\mathbf{Y}$ should not be performed directly from $\mathbf{X}$.

An alternative is to predict $\mathbf{Y}$ from the LVs in $\mathbf{X}$. The aim of the PLS regression is to estimate $\mathbf{Y}$ from the subspace of $\mathbf{X}$ that maximizes its covariance with $\mathbf{Y}$. The partial linear regression problem between normalized matrices $\mathbf{X}$ and $\mathbf{Y}$ can be stated as:

$$\begin{aligned} \mathbf{X} &= \mathbf{T}_A \cdot \mathbf{P}_A^T + \mathbf{E}_A \\ \mathbf{Y} &= \mathbf{T}_A \cdot \mathbf{Q}_A^T + \mathbf{F}_A \end{aligned} \tag{4}$$

where $\mathbf{T}_A$ is the $N \times A$ score matrix that contains the projections of $\mathbf{X}$ to the latent $A$-dimensional subspace, $\mathbf{P}_A$ and $\mathbf{Q}_A$ are the $M \times A$ and $K \times A$ regressor matrices, also called loading matrices, and $\mathbf{E}_A$ and $\mathbf{F}_A$ are the $N \times M$ and $N \times K$ matrices of residuals of $\mathbf{X}$ and $\mathbf{Y}$, respectively. Equation (4) can be rearranged in the following form:

$$\mathbf{Y} = \mathbf{X} \cdot \mathbf{R}_A \cdot \mathbf{Q}_A^T + \mathbf{F} \tag{5}$$

with:

$$\mathbf{R}_A = \mathbf{W}_A \cdot (\mathbf{P}_A^T \cdot \mathbf{W}_A)^{-1} \tag{6}$$

where $\mathbf{W}_A$ is a $M \times A$ matrix of weights. Thus, a PLS model is represented by matrices $\mathbf{P}_A$, $\mathbf{W}_A$ and $\mathbf{Q}_A$.

A variant of PLS for supervised classification is PLS-Discriminant Analysis (PLS-DA) [18]. In PLS-DA, matrix $\mathbf{Y}$ is artificially generated with dummy variables, which codify the different classes in the data set. Typically, $\mathbf{Y}$ is constructed with as many features as classes. All the observations belonging to a class have value 1 for the corresponding feature, and -1 for the rest. This PLS variant is especially useful in the context of data classification, as it will be shown later on in this paper.

Like in PCA, it is customary to mean-center and scale variables in both $\mathbf{X}$ and $\mathbf{Y}$ blocks.

# 3 Exploratory Data analysis

In EDA, PCA and PLS can be used to improve the understanding of a two-way data set, with a number of observations of a number of features, mainly in the following three main aspects:

a) The distribution of the observations, e.g. the distribution of the flows in a traffic data set. This distribution, in particular the existence of outliers and clusters, contains relevant information for data understanding. Thus, outliers represent abnormal situations that may be very informative in certain contexts. Clusters of observations may represent habitual situations. The separation among clusters may reflect different operation points in the phenomenon of interest. Periodical switching among clusters may reflect repetitive patterns.

   Data sets with several tens of features are common in many areas related to networking, for instance in anomaly detection [19] or traffic classification [20]. Because of the dimension of the data, the direct observation of data distribution in these data sets is not possible, and the visualization of selective pairs of features at a time is a tedious approach. PCA and PLS can be used straightforwardly to visualize the distribution of the data in the latent subspace, considering only a few LVs that contain most of the variability of interest. Score plots [9] are used for this purpose.

b) The relationships among features, e.g. is it the mean size of a packet related to the length of the flow?. PCA or PLS can be used to find relationships among features. Traditionally, Factor Analysis (FA) [21] [9] was used for this purpose. Nevertheless, the combination of PCA and PLS projection models with a recent method termed Missing Data Methods for Exploratory Data Analysis (MEDA) [22] outperforms traditional FA techniques. Also, loading plots [9] are useful to investigate the distribution of the features.

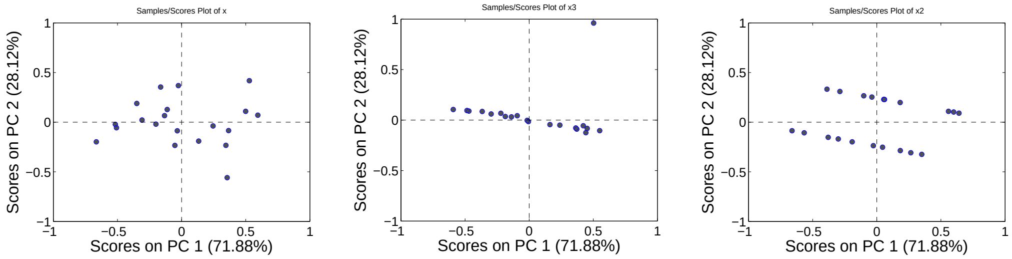
c) The connection between observations and features. The investigation of the connection between observations (e.g. P2P flows) and features (e.g. high mean packet size) in the latent subspace is principal to gain a complete picture of the data, for instance to unveil potential causes for the apparition of outliers and clusters. Traditionally, biplots [23] have been used for this purpose. Also, an extension of MEDA named observation-based MEDA or *o*MEDA [24], is useful in this context.

## 3.1 Score plots

When a projection model is calibrated, no matter the purpose, the distribution of the scores should always be inspected. This is a common flaw in research papers that use PCA for dimension reduction. The distribution of the scores has to be inspected to detect potential artifacts, such as outliers, that may render the dimension reduction inappropriate. The same model (same covariance structure) may be the result of very different data distributions. For instance, in Figure 1, the score plots of three completely different data distributions that provide exactly the same PCA model are shown: a multinormal distribution, a distribution with one outlier and a distribution with two clusters[1]. This example illustrates that the success of the application of projection models requires the correct interpretation of the distribution of the observations. For instance, for dimension reduction, outliers should be isolated from the rest of the data and the model subspace recomputed. Also, for EDA, the three plots represent a very different situation. Coming back to the traffic data example, the first plot represents a normal distribution of flows, the second plot highlights a special traffic flow which could be a network attack, and the third plot shows

[1] The ADICOV algorithm [25] was employed to simulate these data sets.

two differentiated types of traffic, which could be P2P and client-server traffic. Therefore, the inspection of score plots or equivalent tools is strongly recommended.



(a) (b) (c)

Figure 1: Three different scores distributions for the same PCA model with 2 PCs.

## 3.2 MEDA

MEDA [22] is a tool to find the relationships among the features in the data. It can be applied in any subspace of interest, including PCA and PLS. The MEDA approach is depicted in Figure 2. Firstly, a projection model is fitted from the calibration $N \times M$ matrix $\mathbf{X}$ (and optionally $\mathbf{Y}$). Then, for each feature $m$, matrix $\mathbf{X}_m$ is constructed, which is a $N \times M$ matrix full with zeros except in the $m$-th column where it contains the $m$-th column of matrix $\mathbf{X}$. Using $\mathbf{X}_m$ and the model, the scores are estimated with a missing data method. The known data regression (KDR) method [26, 27] is suggested at this point. From the scores, the original data is reconstructed and the estimation error computed. The sum of squares of the estimation error is compared to that of the original data according to the following index of goodness of prediction:

$$q^2_{A,(m,l)} = 1 - \frac{\|\hat{\mathbf{e}}_{A,(l)}\|^2}{\|\mathbf{x}_{(l)}\|^2}, \qquad \forall l \neq m. \tag{7}$$

where $\hat{\mathbf{e}}_{A,(l)}$ corresponds to the estimation error for the $l$-th feature using a model with $A$ LVs and $\mathbf{x}_{(l)}$ is its actual value. The closer the value of the index is to 1, the more related features $m$ and $l$ are. After all the indices corresponding to each pair of features are computed, matrix $\mathbf{Q}^2_A$ is formed so that $q^2_{A,(m,l)}$ is located at row $m$ and column $l$. This matrix is similar in nature to a element-wise squared correlation matrix, but with better capabilities to distinguish between structure and noise [22]. More details on MEDA can be found on [22] and [28].

For interpretation, when the number of features is large, MEDA can be used in combination with loading plots, which are similar to score plots and show the distribution of the features in the model subspace.

## 3.3 *o*MEDA

*o*MEDA [24] is a variant of MEDA to connect observations and features. Basically, *o*MEDA is a MEDA algorithm applied over a combination of the original data and a dummy variable designed to cover the observations of interest. Take the following example: a number of subsets of observations (e.g. traffic flows) $\{\mathbf{C}_1, ..., \mathbf{C}_N\}$ form different clusters in the scores plot that are located far from the bulk of the data, $\mathbf{L}$. One may be interested in identifying, for instance, the features related to the deviation of $\mathbf{C}_1$ from $\mathbf{L}$ without considering the rest of clusters. For that, a dummy variable $\mathbf{d}$ is created so that observations in $\mathbf{C}_1$ are set to 1, observations in $\mathbf{L}$ are set

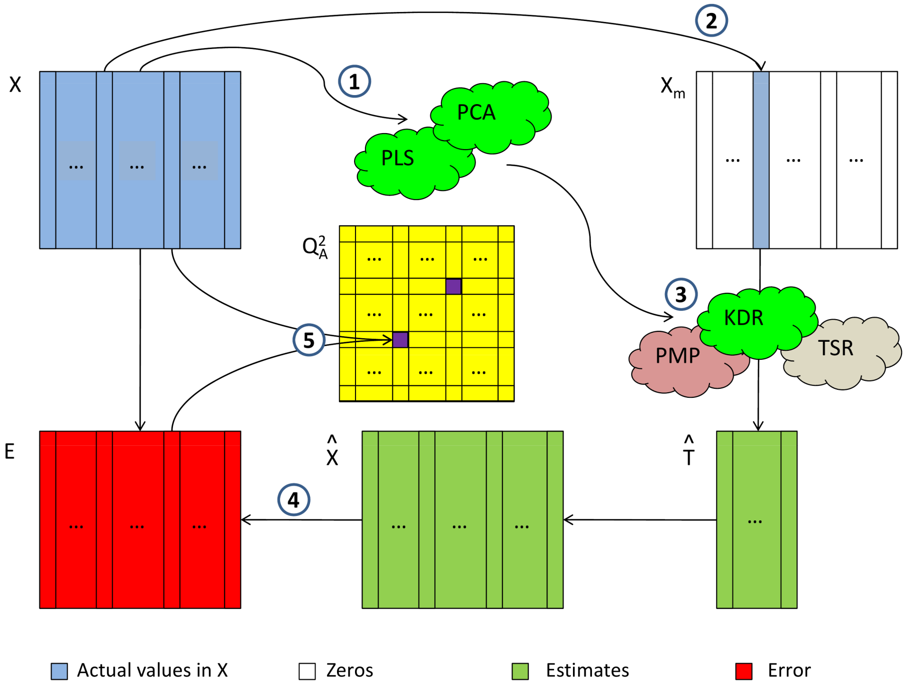


Figure 2: MEDA technique: (1) model calibration: a PCA or PLS model is computed from X; (2) introduction of missing data: to build $\mathbf{X}_m$, all columns in $\mathbf{X}$ except $m$ are filled with zeros; (3) missing data imputation using an algorithm such as Known Data Regression (KDR), Projection to Model Plane (PMP) or Trimmed Score Regression (TSR) [26, 27]; (4) error computation from actual data and estimates; (5) computation of matrix $\mathbf{Q}^2_A$.

to -1, while the remaining observations are left to 0. Also, values (weights) other than 1 and -1 can be included in the dummy variable if desired. *o*MEDA is then performed using this dummy variable.

The *o*MEDA technique works as follows. Firstly, the dummy variable is designed to cover a subset of observations and combined with the data set. Then, a MEDA run is performed by predicting the original features from the dummy variable. The result is a single vector, $\mathbf{d}^2_A$, of dimension $M \times 1$, being $M$ the number of original variables. Values of high magnitude in $\mathbf{d}^2_A$ identify those features related to the deviation among the observations of interest. Also, control limits based on resampling techniques [29] [30] can be obtained. Being $\mathbf{d}$ the dummy variable, the *o*MEDA index follows:

$$d^2_{A,(l)} = \|\mathbf{x}^d_{(l)}\|^2 - \|\hat{\mathbf{e}}^d_{A,(l)}\|^2, \quad \forall l. \tag{8}$$

where $\mathbf{x}^d_{(l)}$ represents the values of the $l$-th variable in the original observations different to 0 in $\mathbf{d}$ and $\hat{\mathbf{e}}^d_{A,(l)}$ is the corresponding estimation error. For a deeper description of the *o*MEDA algorithm refer to [24]. More details on *o*MEDA can also be found on [28].

# 4 EDA in networking data sets of reduced size

This section motivates the use of the EDA framework presented here in networking data sets. It will be shown that the framework can be equally applied to very different data sets, with similar or improved capabilities in comparison with context specific approaches. The obvious benefit is that the IT manager or IT analyst only needs to learn the use of a single visualization approach for handling a wide variety of problems and situations. The tools used in this part of the paper have been included in the EDA Toolbox 2.0 for MATLAB, available online at http://wdb.ugr.es/ jose-camacho/downloads.php

## 4.1 Low-size traffic analysis

The next example illustrates the use of the EDA methodology in the context of traffic analysis [31]. The data analyzed were captured in the Networking Laboratory during a 2 hours laboratory session of the course 'Network Management', in the Telecommunications degree of the University of Granada. During this session, the students were dealing with the configuration of polling and traps generation with the Simple Network Management Protocol (SNMP), in the Cisco routers and switches of the laboratory.

The Networking Laboratory consists of 24 user work stations (noted Px) connected to the different networks configured in the laboratory. These 24 work stations are arranged in cells of 4 stations each. The six cells in the laboratory, numbered from 1 to 6, are able to work autonomously from the rest of the cells. Each cell is composed by 4 work stations (Px/1, Px/2, Px/3 and Px/4, where x stands for the cell number) with three network interfaces each one, connected to three different networks containing a variety of network devices such as three Cisco 1841 routers (Rx-A, Rx-B and Rx-C) and three Catalyst 2950 switches (SWx-A, SWx-B and SWx-C), among others (ATM, Frame Relay and X.25 devices, PBX, etc.).

During part of the laboratory session, SNMP information was acquired from the switches in one of the cells (cell 4) with a one minute sampling interval. The data, acquired with the *snmpwalk* command of the Net-SNMP distribution, were the input and output traffic octets in the 14 interfaces of the three switches. During the session there were two students working in the cell: student 1, located in P4/1, was configuring the SW4-A by means of a Telnet connection, while student 2, located in P4/2, was configuring R4-A, also with a Telnet connection. In addition, during some time intervals in the session, a Neptune attack (SYN flooding) to SW4-C was performed by the teacher, located in P4/4. The experiment was developed so that the students work in the laboratory session was not affected.

The objective of this experiment is to illustrate that the EDA framework can be used to detect anomalies and to identify traffic sources, in particular to perform a forensic analysis with the goal of detecting the source of the attack. The experiment is intentionally simplistic and graphics are annotated in order to simplify the interpretation of EDA for the untrained reader. Alternatively, this type of analysis could be performed using a link graph, where hosts are represented by nodes and connections are represented by edges [5]. Notice that this visualization approach is only aimed at showing interactions between hosts, and cannot be used for most other sources of data in networking. The price to pay using the EDA framework, which is context free, is that the visualization needs to be interpreted taking implicitly the context into consideration. In this example, the interpretation of the EDA tools will be performed taking the topology of the one of the three different networks, named *cell management network* in Figure 3, in mind.

In the experiment, 108 data observations were obtained with 84 variables each. After the data pre-processing to compute counter increments and remove variables with no traffic, 101 observations with 48 variables each one were left in the data set. The list of variables is included

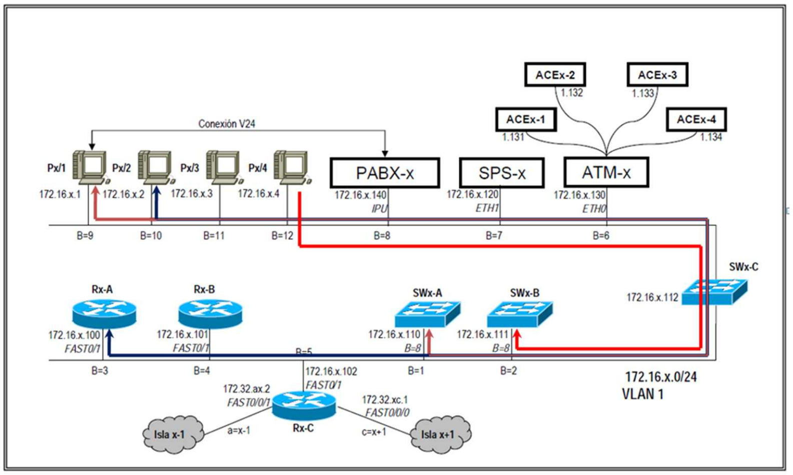


Figure 3: Network connectivity diagram in one of the three networks of the cell (management network) and traffic flow route during the work of both students (student 1 in brown and student 2 in blue) and the Neptune attack (in red). Interface $n$ of SWx-C is represented by the text $B{=}n$ below the horizontal lines. The corresponding interfaces of switches SWx-A and SWx-B connected to SWx-C are also shown ($B{=}8$).

in Table 1. The 101 observations were split into a calibration group, only formed by normal traffic, and a test group, formed by both normal and Neptune attack traffic.

In Figure 4 the MEDA plot of the calibration data based on PCA is shown. Recall that the darker the value of a pair of variables $m$ and $l$, the more related the features are. In the plot, it is possible to discriminate two groups of related variables highlighted in the graph. Notice that the square annotated of lowest size includes also variables of the first group which are not in the second group.

When the MEDA contains many variables, like in the present case, it may be difficult to interpret. To ease the interpretation, MEDA can be combined with a loading plot (Figure 5(a)) which shows the distribution of the variables. Loading plots are more suitable for the visualization of a high number of features. However, unlike MEDA, they are restricted to a 2-dimensional subspace. In a loading plot (and also in a score plot), only objects far from the origin are of interest. Variables near the origin present a low amount of variance in the subspace considered in the plot. Therefore, that subspace is not a valid choice to investigate their variability. Rather, interpretation should be focused on variables (and observations in a score plot) which are far from the origin. Also, a typical interpretation is that variables located close in a loading plot are correlated (and close observations in a score plot are similar) However, it should be noted that this is not always the case and correlation should be confirmed with MEDA. Thus, MEDA and loading plots are complementary visualization graphs which may be combined. This is also true for oMEDA and score plots.

On the other hand, to improve visualization, the MEDA matrix can also be reduced to a set of features of interest and also seriated (reordered according to a similarity criterion)[32]. The result of selecting only the features of both groups detected in Figure 4 (or the loading plot in Figure 5(a)) and obtaining the seriated (reordered) MEDA is shown in Figure 5(b).

The groups of seriated variables found with MEDA and the loading plot are listed in Table 2. A detailed analysis of these variable lets infer and trace the route followed by the network

Table 1: SNMP variables obtained with snmpwalk in the terminals of the three switches of the cell 4.

| # Var | Name | # Var | Name | # Var | Name |
|---|---|---|---|---|---|
| 1 | A.ifIn1 | 18 | B.ifIn9 | 35 | C.ifIn9 |
| 2 | A.ifIn2 | 19 | B.ifIn10 | 36 | C.ifIn10 |
| 3 | A.ifIn8 | 20 | B.ifIn4 | 37 | C.ifIn12 |
| 4 | A.ifIn14 | 21 | B.ifOut1 | 38 | C.ifIn14 |
| 5 | A.ifOut1 | 22 | B.ifOut2 | 39 | C.ifOut1 |
| 6 | A.ifOut2 | 23 | B.ifOut3 | 40 | C.ifOut2 |
| 7 | A.ifOut3 | 24 | B.ifOut4 | 41 | C.ifOut3 |
| 8 | A.ifOut4 | 25 | B.ifOut8 | 42 | C.ifOut4 |
| 9 | A.ifOut8 | 26 | B.ifOut9 | 43 | C.ifOut5 |
| 10 | A.ifOut9 | 27 | B.ifOut10 | 44 | C.ifOut7 |
| 11 | A.ifOut10 | 28 | B.ifOut11 | 45 | C.ifOut9 |
| 12 | A.ifOut11 | 29 | B.ifOut12 | 46 | C.ifOut10 |
| 13 | A.ifOut12 | 30 | B.ifOut14 | 47 | C.ifOut11 |
| 14 | A.ifOut14 | 31 | C.ifIn1 | 48 | C.ifOut12 |
| 15 | B.ifIn1 | 32 | C.ifIn2 | 49 | C.ifOut14 |
| 16 | B.ifIn2 | 33 | C.ifIn3 | | |
| 17 | B.ifIn8 | 34 | C.ifIn5 | | |

Table 2: Groups of variables according to MEDA in Fig. 4. Each group corresponds to the Telnet traffic of each student during the session.

| Group | Variables | Description |
|---|---|---|
| Student 1 | A.ifIn14(4), C.ifIn9(35), C.ifOut1(39), A.ifIn8(3), C.ifOut9(45), A.ifOut8(9), C.ifIn1(31), A.ifOut14(14) | Telnet Configuration of Sw-A from PC-1 |
| Student 2 | C.ifOut10(46), C.ifIn3(35), C.ifOut3(41), C.ifIn10(36) | Telnet configuration of R-A from PC-2 |

traffic in the cell, according to the network diagram of the network topology in Figure 3. The seriation performed with MEDA simplifies this task. In the first group of variables, (input and output) interfaces 8 and 14 of SW4-A are related to interfaces 1 and 9 of SW4-C. The interfaces 8 of SW4-A and 1 of SW4-C are interconnected and the interface 9 of SW4-C is connected to P4/1. Also, interface 14 of SW4-A is a virtual interface which represents VLAN1, the virtual LAN used in the configuration of the management network in Figure 3. The flow of the data can be observed in the seriated variables in Table 2: the first four interfaces are more interrelated, representing the connection from SW4-A to P4/1, while the last four represent the connection from P4/1 to SW4-A. Thus, this network traffic flow corresponds to the Telnet of student 1, between P4/1 and SW4-A. The second group of variables point to the connection of interfaces 3 and 10 of SW4-C. According to the topology, this represents the connection between P4/2 and SW4-A. Also, the incoming and outgoing flows are seen in the seriated variables. Although this is a simplistic example, it illustrates the power of MEDA to aid in the understanding of the traffic in the network.

In Figure 6(a) the score plot of the calibration data is shown. The plot shows two main directions of variability, related to two main sources of traffic. Using oMEDA, these two sources of variability can be explored. To explore the first direction of variability, towards observation 35 in the score plot, a dummy variable $v_d$ is designed where all observations except the following are set to -1:

$$v_d(35) = 2;$$

$$v_d(37, 33, 45) = 1;$$

$$v_d(10, 2, 26, 28, 36, 6, 8, 1, 25, 14, 24, 13, 9) = 0;$$

Thus, observations 35, 37, 33 and 45 are compared to the bulk of the data, and all the obser-

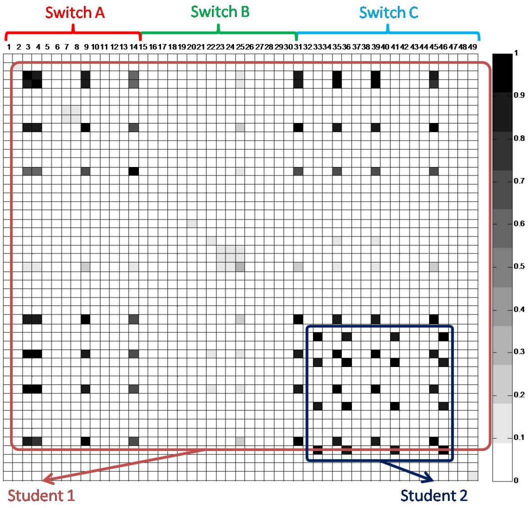
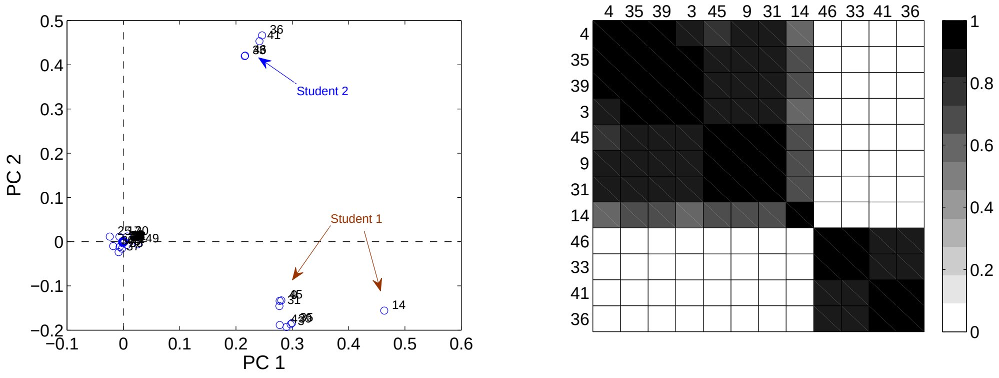


Figure 4: MEDA plot along with the two variability sources originated by the two students work.



(a)

(b)

Figure 5: Loading plot along with the two variability sources originated by the two students work (a) and MEDA matrix after feature selection and seriation (b).

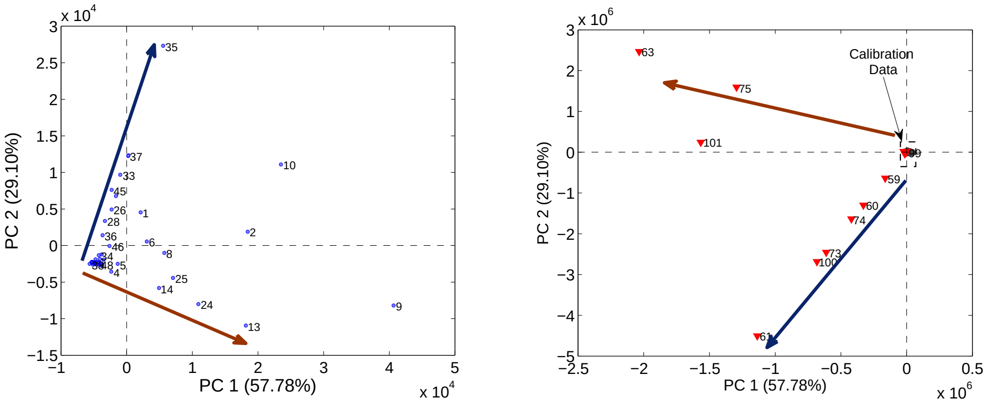
vations which are not in the direction of interest are set to zero, so that they do not contribute to the oMEDA plot. The resulting plot is shown in Figure 7(a). In this plot, the variables related to the traffic of Student 2 (recall Figure 4) are highlighted. This means that as an observation is moving towards the first direction of variability, the proportion of traffic of Student 2 is higher. Thus, observation #35 (upper left corner) represents a minute during which the traffic generated by Student 2 was especially prominent.

The second direction of variability in Figure 6(a) can be studied with the other dummy variable, where all observations are set to -1 except for:

$$v_d(9) = 2;$$

$$v_d(13, 24) = 1;$$

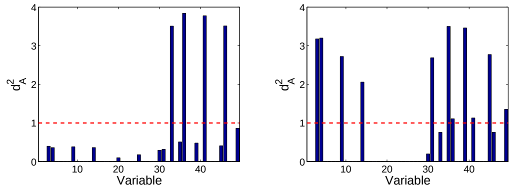
$$v_d(10, 2, 26, 28, 36, 6, 8, 1, 25, 14, 35, 37, 33, 45) = 0;$$



(a)

(b)

Figure 6: Score plot along with the two variability sources originated by the two students work (a) and the Neptune attack (b). The area within the tiny rectangle in the origin of coordinates of (b) approximately corresponds to the area in the whole figure (a).



(a)

(b)

Figure 7: oMEDA plots related to the direction in the score plot towards observation 35 (a) and towards 13 and 9 (b).

The resulting plot is shown in Figure 7(b). Observations towards this direction have a growing amount of traffic of Student 1. Similarly, any type of pattern in the score plot, such as the particular deviation of observation 10, can be straightforwardly studied with oMEDA.

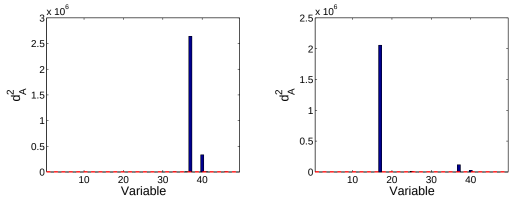
The score plot can be useful to detect anomalies in incoming traffic data. On a first phase, the projection model and score plot of normal data is obtained. Then, new incoming traffic is projected on the model and the new scores are shown in the plot in real-time. If the observations corresponding to new data are far from the calibration data, then some form of anomaly is taking place. To illustrate this, in Figure 6(b) the test observations (including the Neptune attacks) are projected on the score plot of Figure 6(a). A sub-set of the test observations are located further away from the area where the calibration data of Figure 6(a) is located. This sub-set of the test observations corresponds to Neptune attacks in the network. Simply detecting anomalous distances to the calibration data is a means to detect anomalies. The corresponding object residuals should also be inspected. This is essentially the idea beneath PCA-based anomaly detection [19, 33].

Projection models also aid in the diagnosis of the sources of the anomaly, and for this oMEDA is again useful. Thus, in Figure 6(b) two separation directions are observed: from the origin to the observation 61 and from the origin to the observation 63. The corresponding oMEDA plots for those observations are shown in Figure 8. From these plots, the reason for each deviation can be discovered. The first case corresponds to observations at the beginning of the Neptune

Table 3: Traffic statistics in the collected web access.

| Acronym | Description |
|---|---|
| Tv | All the visits - Visits |
| Tp | All the visits - Pages |
| UNv | New Users - Visits |
| UNp | New Users - Pages |
| VRv | Recurrent Visitors - Visits |
| VRp | Recurrent Visitors - Pages |
| TBGv | Free Traffic search engines - Visits |
| TBGp | Free Traffic search engines - Pages |
| TBv | Search traffic - Visits |
| TBp | Search traffic - Pages |
| TDv | Direct traffic - Visits |
| TDp | Direct traffic - Pages |
| TRv | Reference traffic - Visits |
| TRp | Reference traffic - Pages |
| VSRv | Visits without rebound - Visits |
| VSRp | Visits without rebound - Pages |

attacks (observations 59-61, 73-74 and 100), when the packets are forwarded by SW4-C. Thus, the variables highlighted are the input interface 12 of switch C (variable 37, $C.ifIn12$), where P4/4 (the origin of the attack) is connected, and the output interface 2 of switch C (variable 20, $C.ifOut2$), where SW4-B (the destination of the attack) is connected. This plot illustrates the forensics capabilities of the analysis methods presented. The second direction of variability corresponds to the end of the transmission of the Neptune traffic (observations 63, 75 and 101) that mainly affects the input interface in SW4-B (variable #17, $B.ifIn8$). The unidirectional flow of this traffic is also illustrated in Fig. 3.



(a)

(b)

Figure 8: oMEDA plots related to initial (a) and final (b) instants of the Neptune traffic.

## 4.2 Connection statistic in a web page

In this example, the access traffic statistics of the web page of Rekom Biotech S.L. company (http://www.rekombiotech.com) collected during a month period, are analyzed using PCA. In this case, the objective is to identify the trends of access to the web and clues to increment the access of interested clients. The statistics, a total of 16, have been collected with Google Analytics © and are listed in Table 3. The data set is formed by 31 observations, one per day, of 16 variables each one.

The visualization tools of Google Analytics © are limited to time series charts and pie charts, which are univariate or low-dimensional plots at best. For instance, the time series plot of the complete data set, shown in Figure 9, is not an adequate choice for visualization. Using this type of plot, only a reduced number of variables can be properly visualized. Still, the plot shows that

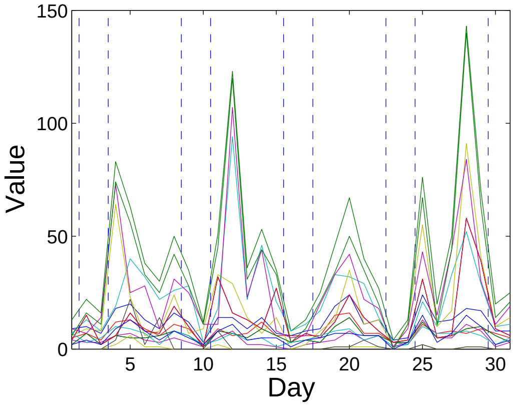
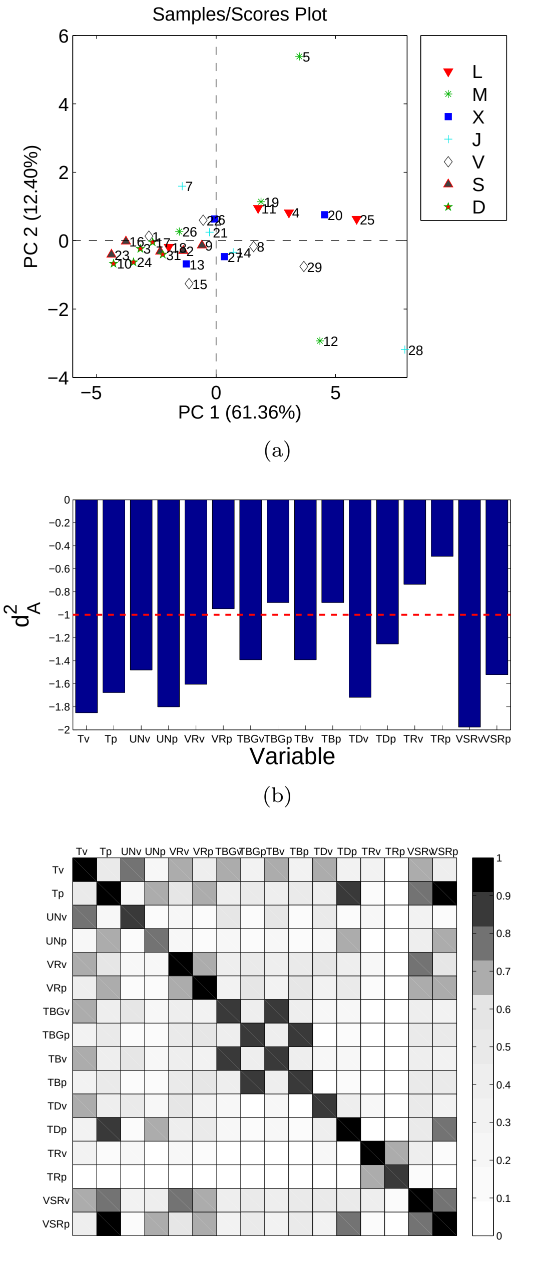


Figure 9: Time series plot of Rekom Biotech web page access data. Weekends and the rest of the week are separated using dashed lines.

the traffic is reduced during weekends and that there are several correlated statistics, which show a similar profile. To investigate such correlation, pairs or thirds of selected variables need to be displayed at a time. This approach is tedious and may lead to wrong conclusions, since all the possible combinations of variables are unlikely to be revised.

In Figure 10, the main visualization tools in the EDA of the web access data are shown. With only those plots, a complete picture of the data is obtained. The score plot (Fig. 10(a)) shows the distribution of the days. The use of different symbols for the days in the week is useful to detect weekly patterns. For instance, weekend days and Fridays are located towards the bottom left corner. This means that there is a different trend of access in working days and weekends. This difference can be investigated with oMEDA. The comparison with oMEDA between weekend and working days, in Fig. 10(b), shows that there is a lower amount of traffic at weekend on a general basis. This reflects the professional interest of the clients in the company. Thus, the web maintenance with down periods should be performed on weekends. The score plot also highlights especial days that are separated from the bulk of the data, for instance the case of day number 5. The traffic of any especial day can be analyzed in detail using oMEDA, in order to identify desired visiting patterns or undesired traffic to be avoided.

The MEDA plot, in Figure 10(c), is the most informative in this example. The first row or column shows that the total of visits (Tv) includes a mixture of new visitors (UNv) and recurrent visitors (VRv). This is seen in the fact that Tv is correlated to UNv and to VRv and at the same time UNv and VRv are not correlated. Thus, the total of Tv is partly UNv and partly VRv. For the same reason, Tv is a mixture of visits coming from search engines (TBGv) and direct traffic (TDv). On the other hand, the number of visited pages (Tp) is also a balanced mixture of new users (UNp) and recurrent visitors (VRp). However, direct visitors (TDp) is correlated to Tp, which reflects that this type of visitors tend to see more pages than people coming from search engines. This is also observed in the correlation of TDp and the visits without rebound (VSRp) An hypothesis on this point is that the reference words in the search engines are not totally adequate, as visitors coming from search engines seem not to be interested in the web page contents. Also, the fact that most of the incoming traffic is direct traffic and that this is partly from newcomers means that the publicity of the web url is being effective. Newcomer may find the url in the professional cards or e-mails of the company staff.



(a)

(b)

(c)

Figure 10: Visualization tools of the EDA of Rekom Biotech web page access data, using PCA: (a) score plot, (b) oMEDA plot of the difference between weekend and the rest of the week and (c) MEDA plot.

## 4.3 Comparative of routing algorithms in MANET

In this third example, a data-set [34] available at the CRAWDAD repository (http://crawdad.cs.dartmouth.edu/) and published in [35] is analyzed. It consists of an outdoor experiment for the comparison of

Table 4: Results provided in [35].

| Routing algorithm | Message delivery ratio | Packets per message | Average number of hops |
|---|---|---|---|
| AODV | 0.50 | 7.50 | 1.61 |
| APRL | 0.20 | 33.30 | 2.11 |
| ODMRP | 0.77 | 45.59 | 2.47 |
| STARA | 0.08 | 150.67 | 1.18 |

three different routing algorithms in a mobile ad-hoc network (MANET) formed by 33 laptops in movement. The evaluated algorithms are: Any Path Routing without Loops (APRL), Ad hoc On-demand Distance Vector (AODV), On-Demand Multicast Routing Protocol (ODMRP) and System- and Traffic-dependent Adaptive Routing Algorithm (STARA).

The experiment was carried out in an athletics court of 225x365 meters, in which each laptop user was moving all over the field in a random manner during one hour and a half, approximately. Each laptop generated low rate traffic during this period of time. Each routing algorithm was used during 15 minutes in disjoint time intervals. The laptops positions were captured via GPS, and registered along with the number of the generated packets (TIN packets), the received ones (TOUT), or the retransmitted ones (SIN and SOUT).

The main results provided in [35] are shown in Table 4. According to the table, ODMRP is the algorithm that attains the highest message delivery ratio and STARA the one attaining the lowest: only 8% of the packets are delivered to the destination. The main reason for this low ratio is argued to be the amount of control packets generated by STARA. This is also noticed from the number of packets per message in that algorithm (second column in the table). On the other hand, AODV is the algorithm that generates the lowest amount of packets per message. Finally, ODMRP and APRL show a high hop ratio in their routes.

In this example, the EDA methodology is used to unveil more details of the experiment. For this purpose, a set of statistics listed in Table 5 are computed from the original data at regular intervals of time, yielding a total of 100 intervals. The design of these statistics is part of the EDA and should be carried out taking into account the investigation goals. Thus, the first 10 variables are related to the distribution and location of the stations (laptops) in the field, while the remaining 8 variables are related to the network traffic. This will help us determine whether some traffic differences are consequence of distribution differences. Among the 100 observations (time intervals), only those in which the four routing algorithms were active are selected (70 observations), and the rest are discarded. Thus, the final data-set is formed by 70 observations on 18 variables.

For the analysis of these data, PLS-DA is applied. The score plot, in Fig. 11, shows that the observations related to each algorithm are easily distinguished with the designed variables. This means that there are significant differences between the algorithms in the data under analysis. Notice that this fact is, by itself, a result that was not reported in Table 4, where only average values are provided with no variability analysis.

In Fig. 12, the differences between the algorithms are studied with oMEDA. Fig. 12(a) provides the comparison between AODV y APRL. Surprisingly, the main differences found are related to the laptop spatial distribution and not to the routing performance. The observations corresponding to APRL present a higher dispersion in the laptop distribution, revealed by a higher value in variables PD, mM and Mm. Also, according to variables n1-n4, the number of laptops closer to the center of mass of the distribution is higher in the observations of AODV. Clearly, under these circumstances, the communication in the observations of AODV is less challenging than in the observations of APRL just because of the location of the laptops, independently of the routing algorithm used. This feature can be also observed when comparing APRL with the rest

Table 5: Considered variables.

| Number | Variable | Description |
|---|---|---|
| 1 | PD | Average distance between laptops |
| 2 | mM | Minimun value for max. distances |
| 3 | Mm | Minimum value for min. distances |
| 4 | cX | X centroid X |
| 5 | cY | Y centroid Y |
| 6 | cZ | Z centroid Z |
| 7 | n1 | Amount of laptops with a distance to the centroid lower than 1/32 of the max. distance |
| 8 | n2 | Amount of laptops with a distance to the centroid between 1/32 and 2/32 of the max. distance |
| 9 | n3 | Amount of laptops with a distance to the centroid between 2/32 and 3/32 of the max. distance |
| 10 | n4 | Amount of laptops with a distance to the centroid higher than 3/32 of the max. distance |
| 11 | nTI | Number of TIN |
| 12 | nTO | Number of TOUT |
| 13 | nSI | Number of SIN |
| 14 | nSO | Number of SOUT |
| 15 | vTI | Volume of TIN |
| 16 | vTO | Volume of TOUT |
| 17 | vSI | Volume of SIN |
| 18 | vSO | Volume of SOUT |

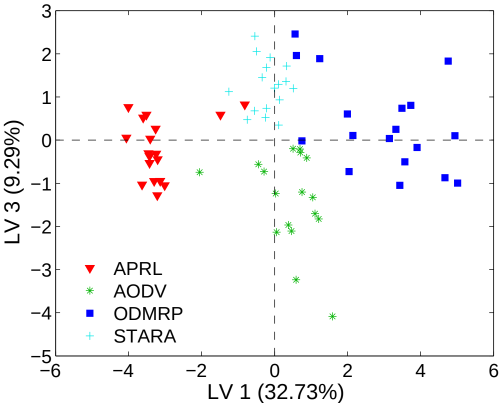


Figure 11: Score plot for the MANET experiment data.

of the routing algorithms (not shown). In this situation, where the dispersion of the stations are significantly bigger for APRL, it is not possible to carry out a reliable comparison with the rest of the algorithms. Stating otherwise, the comparison is not fair because APRL is working on a more complicate scenario than its opponents. Therefore, the results provided in Table 4 for APRL must not be taken into account. This may go unnoticed if the adequate EDA tools are not used. Thus, in [35] it is mentioned that "APRL used longer routes on average than AODV". In addition, the authors state that, although ODMRP tends to emit a high number of packets, "Finally, if we consider the total number of data and control packets versus the total number of messages, we see that ODMRP surprisingly does not fare much worse than APRL". These conclusions are not adequate, considering that the working conditions for APRL are more demanding.

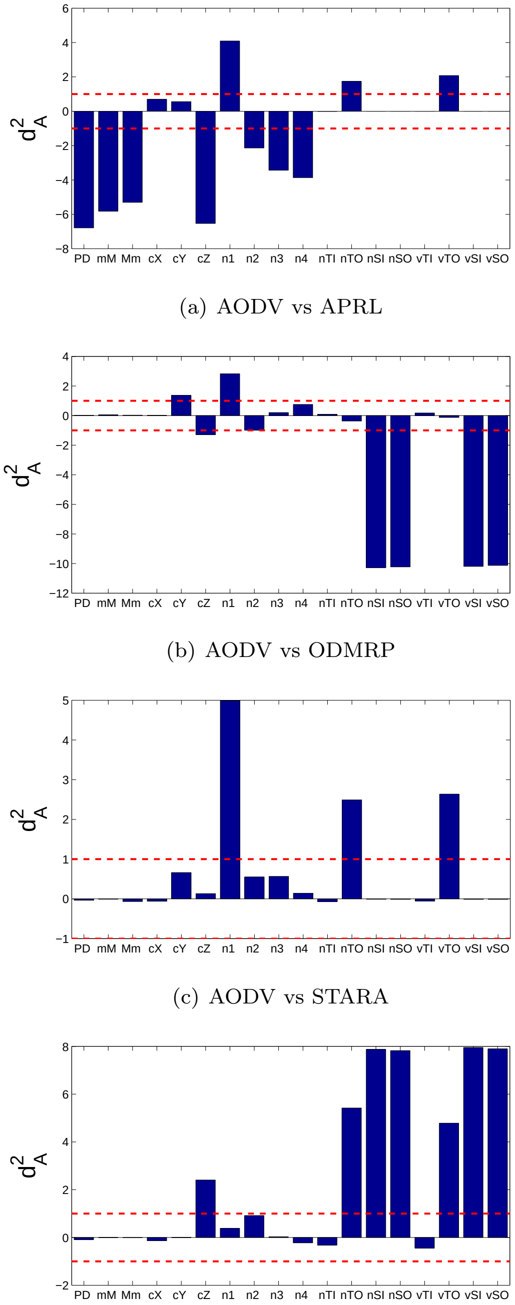
The oMEDA plots between AODV and ODMRP or STARA, Figures 12(b) and 12(c), also show some differences in the distribution of the nodes (e.g. n1), yet there are no differences in the global statistics of the distribution (e.g. PD). Although for an ideal comparative there should not be differences in the distribution of the laptops at all, in this case the comparison seems to be more adequate than the one including APRL. Finally, the comparison between ODMRP and STARA, presented in Figure 12(d), is under ideal conditions. In general terms, it can be seen that STARA implies a low number of packets TOUT, what is coherent with the low message delivery ratio in Table 4. However, the differences between AODV and ODMRP regarding TOUT (or TIN) packets, in Fig. 12(b) seem not to be significant. This fact was confirmed by an adequate hypothesis test: the Welch's test (not shown). The conclusion is that the difference between the message ratio in AODV and ODMRP in Table 4 should not be understood as significant. Another element to be considered, provided by oMEDA, is the high level of retransmitted packets produced by ODMRP. This yields the high number of hops in average, in Table 4.

The clear benefit in the application of the EDA methodology in this example is the better understanding of the multivariate nature in the data. When analyzing data sets using traditional methods, the analyst needs to summarize the variables in a reduced set of statistics. This may obscure the truth underlaying the data, as it was the case in the example. The availability of powerful multivariate tools makes also possible to increment the number of variables, like we did in the example, in order to improve the investigation.

# 5 Extension of the EDA framework for time series data

In the previous section, the use of the visualization tools in the EDA framework based on projection subspaces has been motivated. Using these techniques, a wide class of data sets related to the networking field can be visualized and information can be efficiently extracted. However, there are two common characteristics of networking data that have not been discussed nor addressed. On the one hand, networking data is typically Big data. This means that visualization tools need to be able to handle even millions of time series observations, logs, traffic statistics, etc. The EDA framework presented is suited to handle a very high number of variables, but not for a huge number of observations. On the other hand, networking processes are typically non stationary processes, where the state is continuously changing.

The visibility problem can be overcome using the extension of the framework to Big Data. Refer to [36] [37] for more information. To illustrate this, the data set considered was generated from the 1998 DARPA Intrusion Detection evaluation Program, prepared and managed by MIT Lincoln Labs [38] [39]. The objective of this program was to survey and evaluate research in networking intrusion detection. For that, a large data set including a wide variety of intrusions simulated in a military network environment was provided. The original data set included 4.880.000 observations (connection records). The observations belong to 22 different classes, one class for normal traffic and the remaining for different types of network attacks. Four main categories of attacks were simulated: denial-of-service (DoS), e.g., syn flood; unauthorized access from a remote machine, e.g., guessing password; unauthorized access to local superuser (root) privileges, e.g., buffer overflow attacks; and surveillance and probing, e.g., port-scan. For illustrative purposes, the analysis will be restricted to two types of DoS attack, smurf and neptune, and normal traffic. These three classes represent a 99.3% of the total traffic in the data set. For each connection, 42 features are computed, including numerical and categorical features. To consider categorical features in the



(a) AODV vs APRL

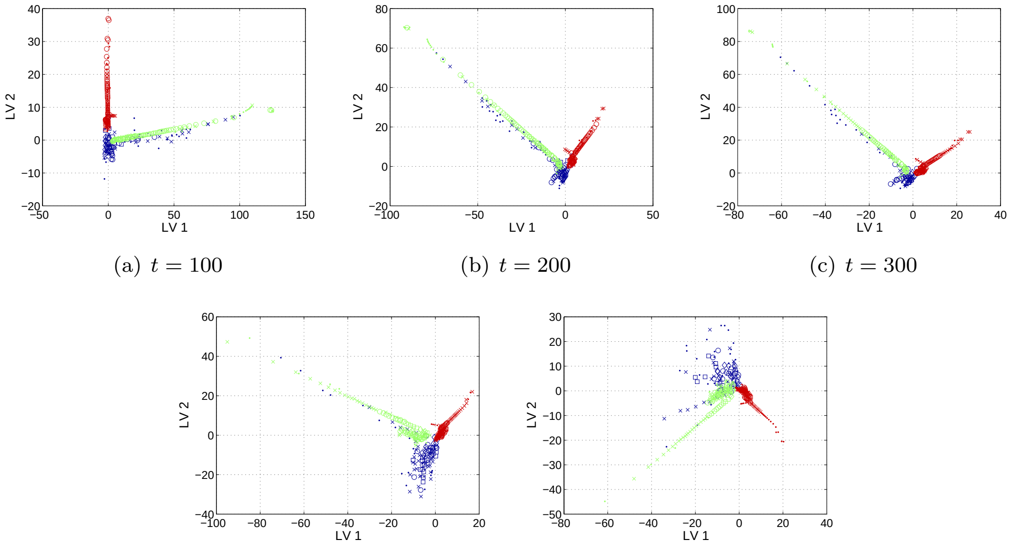
(b) AODV vs ODMRP

(c) AODV vs STARA

(d) ODMRP vs STARA

Figure 12: oMEDA plots for comparison between the routing algorithms.

EDA, one dummy variable per category is included in the data set. The resulting data set contains 4.844.253 observations, each one with 122 features.



(a) $t = 100$

(b) $t = 200$

(c) $t = 300$

(d) $t = 400$

(e) $final$

Figure 13: Compressed score plot for the first two LVs in the PLS-DA model of DARPA data set. The parameter $t$ stands for the number of batch.

The EDA will be aimed at detecting differences between normal traffic and DoS attacks. For that, a PLS-DA model with 2 LVs is chosen. The model is updated for fixed batches of $B = 100$ observations, not representing fixed time periods. Parameter $\lambda$ is set to 0.9999, so that the effective number of observations in the model, $N$, converges to $10^6$. Thus, we have defined the model to be updated slowly so as to capture a long enough past period.

The CSP for the clustering based on Mahalanobis distance in the PLS-DA subspace is shown in Figure 13 for four intermediate points and after the whole data set has been included in the model. Basically, similar conclusions can be derived from the first intermediate point, since the differences among the five models can be basically corrected by a rotation of the axes. Care should be taken to inspect the variability of the LVs in order to interpret the score plot. This can be done by computing the eigenvalues of matrix $\mathbf{W}_A^T \cdot \mathbf{XX} \cdot \mathbf{W}_A$. In the final model, for instance, the variability corresponding to the first LV doubles that of the second LV. Thus, a given distance in the score plot in horizontal direction is two times higher than the same distance in the vertical direction. Considering this information and inspecting the CSP in Figure 13(a), it can be concluded that the set of features provides discriminative capability between both DoS attacks, whereas these attacks can be partially confounded with normal traffic.

In Figure 14, the MEDA colormap computed from the PLS-DA model is shown. This map evidences that only a reduced subset of features is contributing to most of the variance in the data set. Using this information, the number of original features can be reduced from 122 to only 28. This result is of paramount importance for subsequent designs of data mining tools. Figure 15 shows the MEDA of the subset of variables after seriation. In this plot, groups of related features can be identified. The set of selected and reordered features is presented in Table 1.

Using $o$MEDA, the features related to the difference between normal traffic and neptune attacks are identified in Figure 16, and between normal traffic and smurf attacks in Figure 17. The information provided by these plots is valuable both for data understanding and for the hypothet-

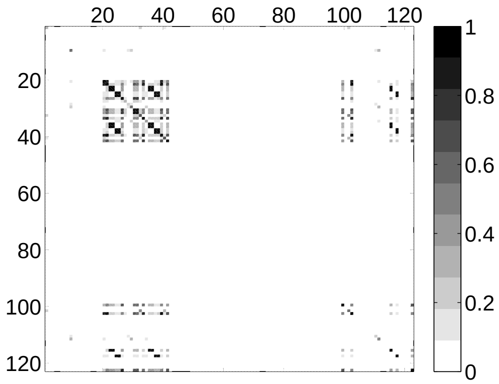


Figure 14: Matrix $\mathbf{Q}_A^2$ by MEDA for the first two LVs in the PLS-DA model of DARPA data set.

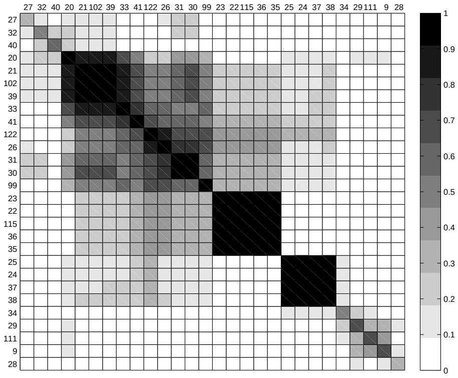


Figure 15: Matrix $\mathbf{Q}_A^2$ by MEDA for the first two LVs in the PLS-DA model of DARPA data set with the reduced set of features.

ical design of traffic classifiers, which should only consider relevant features for the classification. The correctness of the *o*MEDA plot can be assessed by inspecting one of the features of the original data. This is done in Figure 18 for feature 'count'. This plot confirms that, as it was shown in the *o*MEDA plots, smurf attacks show higher 'count' values than normal traffic or neptune attacks. This agrees with which we know from these attacks. Smurf is based on ICMP flooding using broadcast addresses. Thus, it generates a high number of packets. Neptune is based on SYN flooding, which does not need to generate so much packets to cause the denial of service. Notice that, although univariate plots can be useful to validate the *o*MEDA plots, the former cannot show the complex multivariate relationships depicted in *o*MEDA.

The analysis performed on this data set with almost five millions of observations and more than one hundred features, including dummy variables to represent categorical features, illustrates the proposed approach. Only two plots (Figures 13 and 14) were needed to evaluate the discrimination power of considered features and to discard a subset of features that were not useful for discrimination. After variable selection, the model update is performed very efficiently, lets say in

Table 6: Features selected and seriated in the PLS-DA analysis.

| old | new | | |
|---|---|---|---|
| # | # | Name | Type |
| 27 | 1 | | |
| 32 | 2 | dst_host_diff_srv_rate | continuous |
| 40 | 3 | protocol_type: ucp | dummy |
| 20 | 4 | count | continuous |
| 21 | 5 | srv_count | continuous |
| 102 | 6 | service: netbios_ns | dummy |
| 39 | 7 | protocol_type: tcp | dummy |
| 33 | 8 | dst_host_same_src_port_rate | continuous |
| 41 | 9 | protocol_type: icmp | dummy |
| 122 | 10 | flag: SH | dummy |
| 26 | 11 | same_srv_rate | continuous |
| 31 | 12 | dst_host_same_srv_rate | continuous |
| 30 | 13 | dst_host_srv_count | continuous |
| 99 | 14 | service: sql_net | dummy |
| 23 | 15 | srv_serror_rate | continuous |
| 22 | 16 | serror_rate | continuous |
| 115 | 17 | flag: S3 | dummy |
| 36 | 18 | dst_host_srv_serror_rate | continuous |
| 35 | 19 | dst_host_serror_rate | continuous |
| 25 | 20 | srv_rerror_rate | continuous |
| 24 | 21 | rerror_rate | continuous |
| 37 | 22 | dst_host_rerror_rate | continuous |
| 38 | 23 | dst_host_srv_rerror_rate | continuous |
| 34 | 24 | dst_host_srv_diff_host_rate | continuous |
| 29 | 25 | dst_host_count | continuous |
| 111 | 26 | service: red_i | dummy |
| 9 | 27 | logged_in | binary |
| 28 | 28 | | |

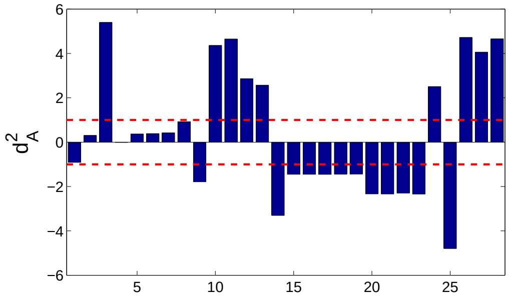


Figure 16: Vector $\mathbf{d}^2_A$ by *o*MEDA to detect the differences between normal traffic and neptune attacks for the first two LVs in the PLS-DA model of DARPA data set with the reduced set of features.

seconds, including the re-computation of cross-product matrices, sums, models, CSPs and MEDA plots. This way, the entire EDA of a large data set can be performed similarly to that of short-scale data sets, without large interruptions corresponding to re-computations.

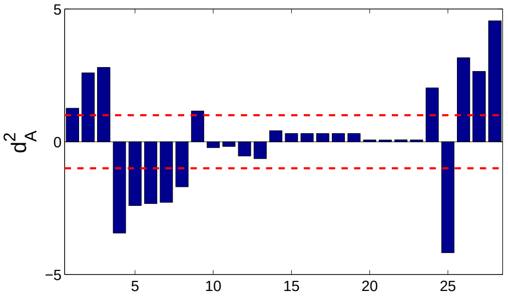


Figure 17: Vector $\mathbf{d}_A^2$ by *o*MEDA to detect the differences between normal traffic and smurf attacks for the first two LVs in the PLS-DA model of DARPA data set with the reduced set of features.

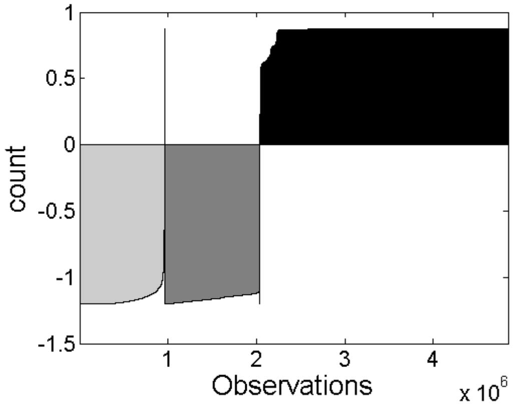


Figure 18: Values of feature 'count' for the three classes: normal traffic, neptune attacks and smurf attacks.

# 6 Conclusion

Exploratory Data Analysis (EDA) means data visualization for understanding. Once the structure of a data set is understood, it is easier to handle it adequately. EDA based on projection models is a very powerful framework to analyze complex data sets with hundreds to thousands of features. A set of recently proposed tools within this framework, revised here, simplifies the interpretation of these complex data sets.

The application of EDA to networking problems has been very limited, specially for large data sets. This is because visualization tools for user-supervision are useless in domains with several thousands or millions of observations. In the networking field there is a clear interest in extending the EDA approach based on projection models for its efficient application in very large data sets. This extension is the contribution of this paper. Combining the inherent capability of projection models to handle large numbers of features with the methods introduced here to deal with large, potentially unlimited, number of observations, the proposed framework is broadly applicable.

The framework is illustrated with several small data sets and a data set including five millions of observations and more than a hundred of features. Results show that with only a few plots correctly interpreted, the user gains a detailed insight into the data, including relationships among observations, relationships among features and crossed relationships. This information, in turn, can be very informative for further data modelling, anomaly detection, diagnosis/forensics, etc.

Concluding, the extended EDA framework proposed here makes the user-supervised analysis of a (big) traffic data set affordable, while avoiding an overwhelming work on the user.

# References

[1] J. Domingo-Pascual, Y. Shavitt, and S. Uhlig, Eds., *TMA'11: Proceedings of the Third international conference on Traffic monitoring and analysis.* Berlin, Heidelberg: Springer-Verlag, 2011.

[2] A. Hafsaoui, G. Urvoy-Keller, D. Collange, M. Siekkinen, and T. En-Najjary, "Understanding the impact of the access technology: the case of web search services," in *Proceedings of the Third international conference on Traffic monitoring and analysis*, ser. TMA'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 37–50.

[3] G. Keren and C. Lewis, *A Handbook for data analysis in the behavioral sciences: statistical issues*, ser. A Handbook for Data Analysis in the Behavioral Sciences. L. Erlbaum, 1993.

[4] G. Ellis and A. Dix, "A taxonomy of clutter reduction for information visualization," *IEEE. Transactions on Visualization and Computer Graphics*, vol. 13, pp. 1216–1223, 2007.

[5] R. Marty, *Applied Security Visualization.* USA: Pearson Education, 2008.

[6] E. Acar and B. Yener, "Unsupervised multiway data analysis: A literature survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, pp. 6–20, 2009.

[7] L. Bernaille, R. Teixeira, and K. Salamatian, "Early application identification," in *Proceedings of the 2006 ACM CoNEXT conference*, ser. CoNEXT '06. New York, NY, USA: ACM, 2006, pp. 6:1–6:12.

[8] C. Dhanjal, S. Gunn, and J. Shawe-Taylor, "Efficient sparse kernel feature extraction based on partial least squares," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 1347–1361, 2009.

[9] J. Jackson, *A user's guide to principal components*, ser. Wiley series in probability and mathematical statistics. Probability and mathematical statistics. Wiley-Interscience, 2003.

[10] S. Wold, "Cross-validatory estimation of the number of components in factor and principal components," *Technometrics*, vol. 20, no. 4, pp. 397–405, 1978.

[11] J. Camacho and A. Ferrer, "Cross-validation in PCA models with the element-wise k-fold (EKF) algorithm: Teoretical aspects," *Submitted to Journal of Chemometrics*, 2011.

[12] Y. Song, F. Nie, C. Zhang, and S. Xiang, "A unified framework for semi-supervised dimensionality reduction," *Pattern Recognition*, vol. 41, pp. 2789–2799, 2008.

[13] X. Zhao and Y. Liu, "Generative tracking of 3D human motion by hierarchical annealed genetic algorithm," *Pattern Recognition*, vol. 41, pp. 2470–2483, 2008.

[14] E. Alaa and D. Hasan, "Face recognition system based on pca and feedforward neural networks," in *Lecture notes in computer science, ISSN 0302-9743*, ser. Lecture Notes in Computer Science, J. Cabestany, A. Prieto, and F. S. Hernández, Eds., vol. 3512. Springer, 2005, pp. 935–942.

[15] H. He and X. Yu, “A comparison of PCA/ICA for data preprocessing in remote sensing imagery classification,” in *MIPPR 2005: Image Analysis Techniques*, D. Li and H. Ma, Eds. Proceedings of the SPIE, Volume 6044, 2005, pp. 60–65.

[16] T. Kourti and J. MacGregor, “Multivariate SPC methods for process and product monitoring,” *Journal of Quality Technology*, vol. 28, no. 4, pp. 409–428, 1996.

[17] A. Ferrer, “Multivariate statistical process control based on principal component analysis (MSPC-PCA): Some reflections and a case study in an autobody assembly process,” *Quality Engineering*, vol. 19, no. 4, pp. 311–325, 2007.

[18] M. Barker and W. Rayens, “Partial least squares for discrimination,” *Journal of Chemometrics*, vol. 17, pp. 166–173, 2003.

[19] D. Brauckhoff, K. Salamatian, and M. May, “Applying pca for traffic anomaly detection: Problems and solutions,” in *INFOCOM 2009. 28th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 19-25 April 2009, Rio de Janeiro, Brazil.* IEEE, 2009, pp. 2866–2870.

[20] T. Nguyen and G. Armitage, “A survey of techniques for internet traffic classification using machine learning,” *Communications Surveys Tutorials, IEEE*, vol. 10, no. 4, pp. 56 –76, quarter 2008.

[21] I. Jolliffe, *Principal component analysis*, ser. Springer series in statistics. Springer-Verlag, 2002.

[22] J. Camacho, “Missing-data theory in the context of exploratory data analysis,” *Chemometrics and Intelligent Laboratory Systems*, vol. 103, pp. 8–18, 2010.

[23] K. Gabriel, “The biplot graphic display of matrices with application to principal component analysis,” *Biometrika*, vol. 58, pp. 453–467, 1971.

[24] J. Camacho, “Observation-based missing data methods for exploratory data analysis to unveil the connection between observations and variables in latent subspace models,” *Journal of Chemometrics*, vol. 25, no. 11, pp. 592–600, 2011.

[25] J. Camacho, P. Padilla, and J. Díaz-Verdejo, “Least-squares approximation of a space distribution for a given covariance and latent sub-space,” *Chemometrics and Intelligent Laboratory Systems*, vol. 105, no. 2, pp. 171 – 180, 2011.

[26] P. Nelson, P. Taylor, and J. MacGregor, “Missing data methods in PCA and PLS: score calculations with incomplete observations,” *Chemometrics and Intelligent Laboratory Systems*, vol. 35, pp. 45–65, 1996.

[27] F. Arteaga and A. Ferrer, “Dealing with missing data in MSPC: several methods, different interpretations, some examples,” *Journal of Chemometrics*, vol. 16, pp. 408–418, 2002.

[28] J. Camacho, *Exploratory Data Analysis using latent subspace models.* INTECH, 2012.

[29] F. Lindgren, B. Hansen, W. Karcher, M. Sjstrm, and L. Eriksson, “Model validation by permutation tests: Applications to variable selection,” *Journal of Chemometrics*, vol. 10, no. 5-6, pp. 521–532, 1996.

[30] S. Wiklund, D. Nilsson, L. Eriksson, M. Sjstrm, S. Wold, and K. Faber, "A randomization test for PLS component selection," *Journal of Chemometrics*, vol. 21, no. 10-11, pp. 427–439, 2007.

[31] "Network forensic frameworks: Survey and research challenges," *Digital Investigation*, vol. 7, no. 12, pp. 14 – 27, 2010.

[32] G. Caraux and S. Pinloche, "Permutmatrix: a graphical environment to arrange gene expression profiles in optimal linear order," 2005.

[33] H. Ringberg, A. Soule, J. Rexford, and C. Diot, "Sensitivity of pca for traffic anomaly detection," *SIGMETRICS Perform. Eval. Rev.*, vol. 35, no. 1, pp. 109–120, Jun. 2007.

[34] R. S. Gray, D. Kotz, C. Newport, N. Dubrovsky, A. Fiske, J. Liu, C. Masone, S. McGrath, and Y. Yuan, "CRAWDAD data set dartmouth/outdoor (v. 2006-11-06)," Downloaded from http://crawdad.cs.dartmouth.edu/dartmouth/outdoor, Nov. 2006.

[35] ——, "Outdoor experimental comparison of four ad hoc routing algorithms," in *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, ser. MSWiM '04, 2004, pp. 220–229.

[36] J. Camacho Páez, G. Maciá Fernández, J. E. Díaz Verdejo, and P. García Teodoro, "Tackling the Big Data 4 Vs for Anomaly Detection," in *INFOCOM'2014 Workshop on Security and Privacy in Big Data*, 2014.

[37] J. Camacho Páez, "Visualizing Big data with Compressed Score Plots: Approach and Research Challenges." *Chemometrics and Intelligent Laboratory Systems*, vol. 135, pp. 110–125, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016974391400080X

[38] "Kdd cup 1999 data set," *The UCI KDD Archive, Information and Computer Science, University of California, http://kdd.ics.uci.edu/databases/kddcup99/kdd cup99.html*, 1999.

[39] H. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, *Selecting features for intrusion detection: a feature relevance analysis on KDD 99 intrusion detection datasets.* Citeseer, 2005, pp. 3–8.