IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 1 June 2022, accepted 15 June 2022, date of publication 24 June 2022, date of current version 30 June 2022.

*Digital Object Identifier 10.1109/ACCESS.2022.3186026*

# Design and Testing Novel One-Class Classifier Based on Polynomial Interpolation With Application to Networking Security

**PIERPAOLO DINI[1], ANDREA BEGNI[1], STEFANO CIAVARELLA[2], EMILIANO DE PAOLI[2], GIUSEPPE FIORELLI[2], CARMELO SILVESTRO[2], AND SERGIO SAPONARA[1], (Senior Member, IEEE)**

[1]Department of Information Engineering, University of Pisa, 56122 Pisa, Italy
[2]MBDA Italia s.p.a, 00131 Rome, Italy

Corresponding authors: Sergio Saponara (sergio.saponara@unipi.it) and Pierpaolo Dini (pierpaolo.dini@phd.unipi.it)

This work was supported by the MBDA Idea2020 Andromaca Project, and in part by the Ministero Istruzione Università Ricerca (MIUR) through the Project "Dipartimenti di Eccellenza Crosslab."

**ABSTRACT** This work exploits the concept of one-class classifier applied to the problem of anomaly detection in communication networks. The article presents the design of an innovative anomaly detection algorithm based on polynomial interpolation technique and statistical analysis. The innovative method is applied to datasets largely used in the scientific community for bench-marking such as KDD99, UNSW-NB15 and CSE-CIC-IDS-2018, and further evaluated with application to a novel available dataset EDGE-IIOTSET 2022. The paper also reports experimental results showing that the proposed methodology outperforms classic one-class classifiers, such as Extreme Learning Machine and Support Vector Machine models, and rule-based intrusion detection system like SNORT. With respect to binary classifiers, this work has the advantage of not requiring any a-priori knowledge about attacks and is based on the collection of only normal data traffic.

**INDEX TERMS** Anomaly and intrusion detection, machine learning, statistical learning theory, classification, data management, networking.

## I. INTRODUCTION

### A. MOTIVATIONS

For the development of Anomaly and Intrusion Detection Systems (ADS ad IDS) there is a growing interest in the use of Machine Learning (ML) and Artificial Intelligence (AI) concepts. Some recent works as [1]–[3] use the ML and AI-based methodologies to explore the various ways of detecting malicious attacks in the computer's networks. In other works in literature for cybercrime [4]–[7], it has been already demonstrated that ML and AI-based methodologies have the potentiality to outperform rules-based IDS tools, such as SNORT and WIRE-SHARK. This is mainly due to the flexibility of ML/AI models. In fact, rules-based algorithms need a very deep knowledge about attacks mechanism in order to elaborate a specific recognition path. This represents a non flexible design process since it is required to define a rule for each of the possible anomalies. This is unfeasible because the number of attack classes grows every day. Instead, ML/AI-based models do not require too much knowledge about the attack and its mechanisms since based on collected data the attacks can be grouped, characterized and then recognized by a ML/AI algorithm. The process of learning from data is very flexible with respect to classic rules-based approach. Indeed, if the condition change the model can be re-trained on new data in order to recognize a new class of attack. A major limit to the ML/AI-based design process is the a priori knowledge about the relationship between the attack classes and the collected data observations. In many applications this represents a practical limit since it could be hard to collect anomalous data traffic, because of the impossibility to replicate some attack classes. This is often due to the non specific knowledge of system designer about the possible attacks. To overcome this limit a new approach based on one-class classifier is proposed in this paper. Indeed a one-class classifier [8] does not require any

The associate editor coordinating the review of this manuscript and approving it for publication was Firooz B. Saghezchi.

knowledge about attacks and is based on collection of only normal data traffic in the communication system of interest. The normal traffic is characterized and is elaborated by a threshold-based logic to determine if the next observations are normal or anomalous.

### B. RELATED WORKS

As anticipated, most of the works in which an IDS based on ML/AI techniques is presented, exploit supervised learning paradigms, where a priori knowledge about the anomalous behaviour and the specific anomaly types is required. In addition, many works limit themselves to testing their algorithms on a single dataset, limiting the validity of the achieved results. For example, in [4], the authors mainly analysed the problem of binary classification on a single dataset. This obviously requires a priori knowledge of anomalous and normal traffic. This problem can be solved by means of one-class classification algorithms. The one-class method is also robust to new type of attacks. In [5]–[7] the use of ML models is presented and compared in terms of performance, but with reference only to the use of the KDD99 dataset (or modified versions). However, there is no contribution to the development of innovative techniques but simply the use of existing models combined with known data manipulation/ reduction techniques. The most critical points, however, are the failure to compare these models on different types of dataset, which is in fact also the only way to verify the validity of the performance analyses and the need to draw on an already labelled dataset to recognise the various types of attack. Similarly, in [9]–[13] supervised learning of known models in the literature is used, where minor modifications in the numerical optimisation algorithms are proposed, referring only to the UNSW-NB15 dataset for the evaluation of the obtained performance. Similar arguments apply to other works in the recent literature, such as [14]–[16], in which the authors present results related to the performance of proposed methods or classical ML/AI models considering only one type of dataset such as the CSE-CIC-IDS-2018. The great limitation of the proposed techniques lies in the fact that a priori knowledge of the types of threats that can affect the computer network is required. This knowledge is not always easy to access. Furthermore, previous Anomaly and Intrusion Detection approaches [4]–[7], [9]–[16] can be easily bypassed by new attack techniques. Few works propose the use of one-class models, and even fewer propose a comparison across multiple datasets, as we propose in this paper. For example, in [17], [18] the use of Extreme Learning Machines (ELM) is proposed as an alternative to Auto-Encoders based on artificial neural networks, with the aim of decreasing learning times, memory requirements for saving weights in memory and computational complexity, in the sense of waiting times during the processing of new observations. The main problem with ELM models is that the transformation matrix is based on random processes that often do not apply, with a strong dependency on the analysed dataset. In literature there are also works [19], [20] proposing the use of Support Vector Machines (SVM) in a one-class version. However, these SVM-based works are characterized by higher waiting times and often modest performance in terms of False Positive Rate.

### C. CONTRIBUTIONS

To overcome the limits of the state of art this paper proposes an anomaly detection technique based on the concept of pre-processing features reduction, polynomial interpolation and one-class model that needs only normal behaviour data.

The contributions of this work are the following:

- design of an innovative one-class technique based on numerical computing algorithms combined with statistical analysis and machine learning.
- higher performance than other one-class techniques in the literature, tested on the three most commonly used datasets overall and on a very recent dataset representative of IIoT applications.
- accuracy essentially in line with results reported in the literature where binary classifiers are used, highlighting the most important strength of the proposed method, related to the non-need for a priori knowledge in terms of collected observations of anomalous behavior.
- exhaustive analysis of the algorithm's robustness and independence from the dataset on which it is applied, proposing the test on the KDD99, UNSW-NB15, CSE-CIC-IDS 2018 and EDGE-IIOTSET 2022 datasets.

### D. PAPER ORGANIZATION

The paper is organized as follows. Section II reviews the selected datasets (KDD99, UNSW-NB15, CSE-CIC-IDS-2018 and EDGE-IIOTSET 2022) used for the design and verification phase and to assess the portability of the proposed technique in different scenarios. Section III describes the proposed algorithm which includes multiple steps such as preliminary data-set manipulation, application of features reduction techniques, polynomial interpolation for normal behavior recognition and final one-class decision policy. Section IV presents the achieved results when applying the novel proposed to the selected datasets. In Section V we report a discussion on the obtained performance, proposing an interpretation based on quantitative and graphical results. Finally, in Section VI we report the conclusion on the proposed novel approach and considerations on future works.

## II. A BRIEF DESCRIPTION OF THE SELECTED DATASETS

In this section we reports a brief description about the dataset on which the proposed method is applied. The choice of the selected datasets derive from a preliminary study of the state of the art on IDS issues, reveling that KDD99, UNSW-NB15 and CSE-CIC-IDS are the most important benchmarks for evaluating new algorithms [21]–[29]. We also propose to apply the innovative proposed methodology to a novel dataset generated and released in 2022, representative of IIoT applications.

### *A. KDD99*

The Kaggle version of the KDD Cup 99 [30], [31] is available an online dataset. The dataset is composed by a total of 25192 TCP/IP connections (observations) from a simulated typical US Air Force LAN. The competition task was to build a network intrusion detector, a predictive model capable of distinguishing between bad connections, called intrusions or attacks, and good normal connections. For this reason each connection is labelled as either normal or anomalous. Each connection record consists of about 100 bytes. For each TCP/IP connection, 38 quantitative and 4 qualitative features are obtained for a total of 42 features. The dataset size 25192 × 42, i.e. more than 1 million of elements.

### *B. UNSW-NB15*

The UNSW-NB15 dataset [32] is available online dataset, free to use and globally acknowledged as a valid benchmark for testing intrusion detection systems. The UNSW-NB15 dataset was created by the IXIA PerfectStorm tool in the Cyber Range Lab of Canberra University for generating a hybrid of real modern normal activities and synthetic contemporary attack behaviours. The tcpdump tool was utilised to capture 100 GB of the raw traffic. This dataset presents 9 types of attacks. The Argus, Bro-IDS tools are used and twelve algorithms are developed to generate totally 49 features. The total number of records in the dataset is 2540047, split in 4 .csv files. In the repository, besides the .csv files, it is possible to find also raw traffic records as .pcap files. The dataset size is 2540049 × 49, i.e. about 125 millions 0f elements.

### *C. CSE-CIC-IDS-2018*

The CSE-CIC-IDS-2018 is an online available dataset widely used in literature for testing and evaluation the performance of ADS/IDS crated by University of New Brunswick. It includes seven different attack scenarios: Brute-force, Heartbleed, Botnet, DoS, DDoS (Distributed DoS), Web attacks, and infiltration of the network from inside [33], [34]. The attacking infrastructure includes 50 machines and the victim organization has 5 departments and includes 420 machines and 30 servers. The dataset includes the captures network traffic and system logs of each machine, several features extracted from the captured traffic using CICFlowMeter-V3. In CSE-CIC-IDS 2018 dataset, the authors use the notion of profiles to generate datasets in a systematic manner, which will contain detailed descriptions of intrusions. The dataset size is 16233002 × 80, i.e. about 1.3 billions of elements.

### *D. EDGE-IIOTSET 2022*

The EDGE-IIOTSET 2022 is new available online dataset [35]. In this work a new IoT and IIoT dataset collected from seven-layer tested including more than 10 IoT devices, IIoT-based Modbus flows, 14 IoT and IIoT protocol-related attacks, are proposed. The dataset traffic is generated by various IoT devices such as Low-cost digital sensors for sensing temperature and humidity, Ultrasonic sensor, Water level detection sensor, pH Sensor Meter, Soil Moisture sensor, Heart Rate Sensor, Flame Sensor, etc.. The authors propose fourteen different type of attacks which are categorized into five threats: DoS/DDoS attacks, Information gathering, Man in the middle attacks, Injection attacks and Malware attacks. The authors provide a reduced version of the entire dataset in order to test ML methods. The dataset size is 157800 × 63, i.e. about 10 millions of elements.

## III. POLYNOMIAL-BASED ONE-CLASS CLASSIFIER DESIGN

### *A. METHODOLOGY DESCRIPTION*

Our approach is based on the idea of being able to extract the polynomial features from the dataset of normal traffic observations. In that way, it's possible to characterize an area of normal behaviour of the system based on the polynomial interpolation over features used as a training dataset. The normality area is then defined as the bounded area of the upper and lower extremes of the polynomials belonging to the training datset. Once the normality area has been defined, it is safe to assume that the polynomials of the normal traffic data not belonging to the training dataset are contained within the normality area. This means that most (if not all) of the points of the polynomial extracted from a new normal traffic observation belong to the area extracted in the training phase. At the same time, an observation of abnormal traffic (e.g. an attack) must be difficult to overlap with the area of normal observations. This must mean that the number of points of the polynomial extracted from the attack that lie within the normal area will be noticeably smaller than the points of the polynomials extracted from the anomalous traffic. Thus, by defining an anomaly threshold, it is possible to distinguish whether a new observation is an attack or not. Using this kind of paradigm, it is not necessary to know all kind of traffic types but it is enough to collect just the normal observation to develop a one-class classifier. The various steps of the algorithm are developed and will be described in detail in the following sections.

### *B. PRELIMINARY DATASETS MANIPULATION*

Independently on the dataset to which our workflow is applied, some manipulation operations of the feature values are performed, in order to make the subsequent steps easier. One of the first manipulations consists in assigning numerical values to features made up of symbolic values, such as the Timestamp or the number of the communication port. In the proposed method, the remapping of variables, of the "Label Encoding" type [36], has been adopted. The reason for this choice is to keep the number of features in the datasets as low as possible at each operational step. Another necessary manipulation relates to reducing the likelihood of bad calculations. In particular, the "min-max" normalisation procedure is applied to the columns of the dataset, as shown in Equation. 1.

$$f_i = \frac{f_i - min\left(\vec{f}\right)}{max\left(\vec{f}\right) - min\left(\vec{f}\right)} \tag{1}$$

where $\vec{f}$ denotes the starting column vector and $f_i$ the component $i^{th}$, whose value is reassigned according to the definition given. Following the normalisation of the feature values, observations in which NaNs are present, which have no information content, are removed.

### C. APPLYING FEATURES REDUCTION

In order to preliminarily reduce the number of features in the datasets, the PCA (Principal Component Analysis) [37] and MDS (Multi-Dimensional Scaling) [38] techniques are applied simultaneously during the dataset preparation and learning phases.

#### 1) PRINCIPAL COMPONENT ANALYSIS

Starting from the original dataset in matrix form, as reported in the following.

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{Nn} \end{bmatrix} = \left[\vec{A}_1 | \vec{A}_2 | \cdots | \vec{A}_n\right] \in \mathbb{R}^{N \times n}$$

where $\vec{A}_k$ is the $k^t h$ column vector of the matrix $A$, for each of the dataset column it is computed the mean value, organized in a column vector that must be subtracted to the $\vec{A}_k$ itself.

$$\begin{aligned} \mu_1 &= \sum_{i=1}^{N} \frac{a_{i1}}{N} = \sum_{i=1}^{N} \frac{[\vec{A}_1]_i}{N} \longrightarrow \vec{\mu}_1 = \mu_1 \vec{1}_N \\ \mu_2 &= \sum_{i=1}^{N} \frac{a_{i2}}{N} = \sum_{i=1}^{N} \frac{[\vec{A}_2]_i}{N} \longrightarrow \vec{\mu}_2 = \mu_2 \vec{1}_N \\ &\cdots \\ \mu_n &= \sum_{i=1}^{N} \frac{a_{in}}{N} = \sum_{i=1}^{N} \frac{[\vec{A}_n]_i}{N} \longrightarrow \vec{\mu}_n = \mu_n \vec{1}_N \end{aligned}$$

Then is defined the matrix $B$, that is basically the matrix $A$, in which in each column is element-wise subtracted the mean value. This facilitates the computation of the co-variance matrix.

$$B = \left[\vec{A}_1 - \vec{\mu}_1 | \vec{A}_2 - \vec{\mu}_2 | \cdots | \vec{A}_n - \vec{\mu}_n\right]$$

The co-variance matrix of the original dataset $A$ can be defined in terms of the $B$ matrix columns as reported hereafter.

$$\Theta = \begin{bmatrix} \Theta_{11} & \Theta_{12} & \cdots & \Theta_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ \Theta_{n1} & \Theta_{n2} & \cdots & \Theta_{nn} \end{bmatrix}$$

where each component of the co-variance matrix has the following form.

$$\Theta_{ij} = \left(\vec{A}_i - \vec{\mu}_i\right) \cdot \left(\vec{A}_j - \vec{\mu}_j\right)$$

Once the co-variance matrix it is computed, it is needed to derive eigenvalues and eigenvectors, in order to identify new features representation space and select the so-called principal components. The principal components are defined as the minimum features that contain most of the information with respect to the original dataset.

$$\begin{aligned} det\left([\Theta - \lambda I_n]\right) &= 0 \longrightarrow \vec{\lambda} = [\lambda_1 \; \lambda_2 \; \cdots \; \lambda_n]^T \\ \tilde{\vec{\lambda}} &= sort\left(\vec{\lambda}\right) \\ \left[A - \tilde{\lambda}_k I_n\right] \vec{v}_k &= \vec{0} \longrightarrow V = \left[\vec{v}_1 | \vec{v}_2 | \cdots | \vec{v}_n\right] \in \mathbb{R}^{n \times n} \\ \tilde{V} &= V(:, 1:m) = \left[\vec{v}_1 | \cdots | \vec{v}_m\right] \in \mathbb{R}^{n \times m} \end{aligned}$$

The new representation space is derived by multiply the matrix $A$ with the eigenvectors matrix, as below.

$$A_{new} = A\tilde{V} \in \mathbb{R}^{N \times m}$$

#### 2) MULTI-DIMENSIONAL SCALING

Multi-Dimensional Scaling is an alternative feature transformation/reduction technique, based on different metrics with respect to PCA. The starting point is the same, the original dataset in matrix form $A$. The "similarity" matrix it is defined as follow.

$$\begin{aligned} D &= \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nn} \end{bmatrix} \\ &= \begin{bmatrix} \|\vec{A}_1 - \vec{A}_1\|^2 & \|\vec{A}_1 - \vec{A}_2\|^2 & \cdots & \|\vec{A}_1 - \vec{A}_n\|^2 \\ \|\vec{A}_2 - \vec{A}_1\|^2 & \|\vec{A}_2 - \vec{A}_2\|^2 & \cdots & \|\vec{A}_2 - \vec{A}_n\|^2 \\ \vdots & \vdots & \ddots & \vdots \\ \|\vec{A}_n - \vec{A}_1\|^2 & \|\vec{A}_n - \vec{A}_2\|^2 & \cdots & \|\vec{A}_n - \vec{A}_n\|^2 \end{bmatrix} \end{aligned}$$

It is defined also a "double centering" matrix $C$, that will be useful to build the matrix for the change of features representation space.

$$C = I_n - \frac{1}{n}\vec{1}_n\vec{1}_n^T = \begin{bmatrix} \frac{n-1}{n} & -\frac{1}{n} & \cdots & -\frac{1}{n} \\ -\frac{1}{n} & \frac{n-1}{n} & \cdots & -\frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{1}{n} & \cdots & -\frac{1}{n} & \frac{n-1}{n} \end{bmatrix}$$

Then is defined the "barycenter matrix" $B$ applying the following congruence linear transformation to the similarity matrix. The eigenvalues and eigenvectors of such matrix are computed, in order to derive the matrix for the features representation space changing $T$.

$$\begin{aligned} B &= -\frac{1}{2}C^T DC \longrightarrow \lambda_1 \; \cdots \; \lambda_n \\ &\longrightarrow det[B - \lambda_k I_n]\vec{t}_k = 0 \longrightarrow T = [\vec{t}_1 | \cdots | \vec{t}_m] \end{aligned}$$

In particular the matrix $T$ is the collection of the eigenvectors related to the $m$ eigenvalues with higher values.

The dataset represented in the new features space can be obtained with the following computation.

$$A_{new} = AT\Delta = \left[\vec{A}_1 | \cdots | \vec{A}_n\right] \left[\vec{t}_1 | \cdots | \vec{t}_m\right] diag\left(\lambda_1, \cdots, \lambda_m\right)$$

### D. POLYNOMIAL INTERPOLATION FOR NORMAL BEHAVIOUR RECOGNITION

Following the procedure for reducing the number of features, based on one of the two methods described above, we propose the use of a polynomial interpolation technique as a further manipulation of the data and to define a decision criterion. In particular, polynomial interpolation is proposed, in our workflow, as an additional criterion for feature reduction and transformation, so as to introduce a further degree of uniqueness for normal behaviour. The degree of the chosen polynomial is less than the cardinality of the dataset in the face of manipulation procedures (normalisation and elimination of observations with NaN) and reduction by transformation of the representation space (via PCA or MDS). As a polynomial interpolation technique we use the least squares criterion with equally spaced interpolation nodes [39]. In Eq. 2 is reported briefly the Least Mean Squares methods from which the polynomial coefficients are derived.

$$\begin{aligned} J^{(i)} &= \sum\nolimits_{h=1}^{n} |p^{(i)}(h) - f_h^{(i)}|^2 \\ \vec{\nabla} J^{(i)} &= \vec{0} \longrightarrow \left[a_0^{(i)} \ \cdots \ a_m^{(i)}\right]^T \end{aligned} \tag{2}$$

For each observation $i^{th}$, and for each coefficient the confidence interval is calculated, as reported in the set of expressions in Eq.3.

$$\begin{aligned} \mu_{a_0} &= \sum\nolimits_{i=1}^{N} \frac{a_0^{(i)}}{N} \quad s_{a_0} = \sum\nolimits_{i=1}^{N} \frac{\left(a_0^{(i)} - \mu_{a_0}\right)^2}{N-1} \\ &\vdots \\ \mu_{a_m} &= \sum\nolimits_{i=1}^{N} \frac{a_m^{(i)}}{N} \quad s_{a_m} = \sum\nolimits_{i=1}^{N} \frac{\left(a_m^{(i)} - \mu_{a_m}\right)^2}{N-1} \end{aligned} \tag{3}$$

Thus the upper and lower 'boundary' polynomials coefficients can be defined as described in Eq.4.

$$\begin{aligned} \vec{a}_{upper} &= \left[\mu_{a_0} + 3\, s_{a_0} \ \mu_{a_1} + 3\, s_{a_1} \ \cdots \ \mu_{a_m} + 3\, s_{a_m}\right]^T \\ \vec{a}_{lower} &= \left[\mu_{a_0} - 3\, s_{a_0} \ \mu_{a_1} - 3\, s_{a_1} \ \cdots \ \mu_{a_m} - 3\, s_{a_m}\right]^T \end{aligned} \tag{4}$$

Then are defined the upper and lower polynomial curves of the normality bound, as reported in Eq.5

$$\begin{aligned} p_{upper}(f) &= \sum\nolimits_{k=0}^{m} \left(\mu_{a_k} + 3s_{a_k}\right) f^k \\ p_{lower}(f) &= \sum\nolimits_{k=0}^{m} \left(\mu_{a_k} - 3s_{a_k}\right) f^k \end{aligned} \tag{5}$$

To decide if an observation is an anomaly in the polynomial representation domain, we evaluate the total number of points that the polynomial curve related to the current observation remains out of the bound, along the entire interval $f_1, \ldots, f_n$. In order to calculate the number of times the polynomial for the i-th observation remains outside the normality interval, it is sufficient to compare the polynomial with the normality bundle itself, in the definition interval.

### E. DECISION POLICY

The decision process we propose is based on the comparison between the interpolating polynomial associated with a new observation to be analysed (in fact we talk about the "on-line" phase) $p_{obs}(x)$ and the normality limits $p_{upper}(x), p_{lower}(x)$. In particular, the number of times $p_{upper}(x)$ is out of normal limits is evaluated. The threshold for applying this decision criterion is evaluated in the off-line phase, in order to derive the 100% performance on the portion of the dataset used for the construction of $p_{upper}(x)$ and $p_{lower}(x)$. Note that this threshold depends on the dataset under consideration. The evaluation of the values assumed by the polynomials is obviously done within the range of features derived from the PCA/MDS procedure, where the index of each feature is also an interpolation node.

In symbols, given $\vec{f}_{new} = [v_1, v_2, \ldots, v_n]$ the vector containing the values of the features in the initial representation base, this is processed by means of the PCA or MDS technique, which as described above is basically equivalent to a multiplication with a matrix for the change of representation base. Given $M_T \in \mathrm{R}^{n \times m}$ the transformation matrix, we obtain a transformed observation $\hat{\vec{f}}_{new} = \vec{f}_{new} M_T = [\hat{v}_1, \hat{v}_2, \ldots \hat{v}_m]$ with $m \leq n$. The components of the vector $\hat{\vec{f}}$ are used to derive the interpolating polynomial $p_{new\_obs} = a_{0,new} + a_{1,new} f + \ldots + a_{m,new} f^m$, which is eventually processed by the function implementing the decision logic.

The decision-making policy can be summarised with the pseudo-code representation shown in List 3.

### F. PERFORMANCE INDEXES

For the evaluation of the models detection performance we use classic index of the ROC (Receiver Operating Characteristic) analysis [40], as reported in the following.

- True Positive (TP) it is an outcome where the model correctly predicts the positive class
- True Negative (TN) it is an outcome where the model correctly predicts the negative class
- False Negative (FN) it is an outcome where the model incorrectly predicts the positive class
- False Positive (FP) it is an outcome where the model incorrectly predicts the negative class

Detection are evaluated through the following quantity.

- Detection rate DR (also identified as "true positive rate", "recall", "sensitivity") is the proportion of attacks that are correctly detected

$$DR = \frac{TP}{TP + FN}$$

- False positive rate FPR (or "false alarm rate") is the proportion of normal traffic incorrectly flagged as attack

$$FPR = \frac{FP}{TN + FP}$$

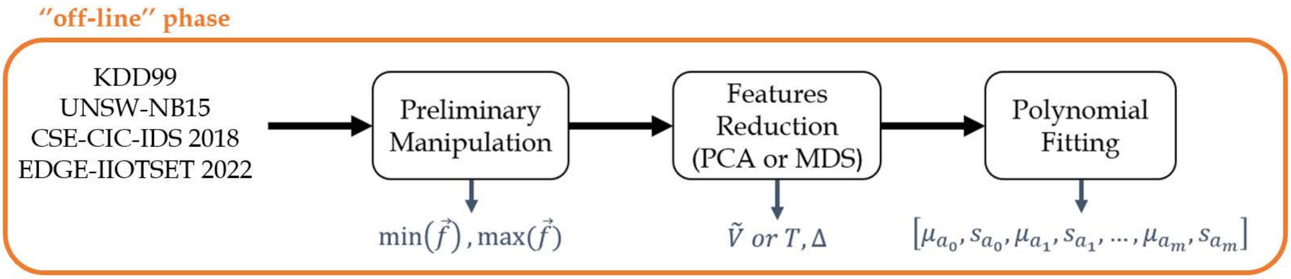
"off-line" phase
KDD99
UNSW-NB15
CSE-CIC-IDS 2018
EDGE-IIOTSET 2022
Preliminary Manipulation
Features Reduction (PCA or MDS)
Polynomial Fitting
$\min(\vec{f}), \max(\vec{f})$
$\tilde{V}$ or $T, \Delta$
$[\mu_{a_0}, s_{a_0}, \mu_{a_1}, s_{a_1}, \dots, \mu_{a_m}, s_{a_m}]$

**FIGURE 1.** Steps of the "off-line" phase in proposed workflow.

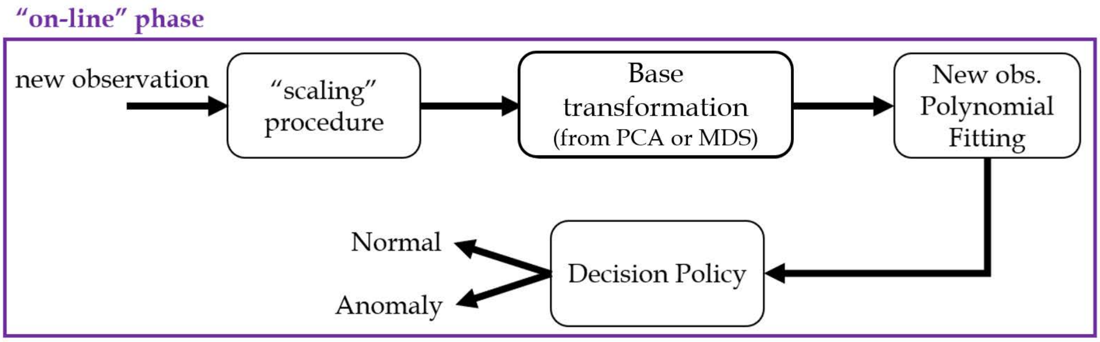

**FIGURE 2.** Steps of the "on-line" phase in proposed workflow.

```
anomalous_counter = 0;
for i in features_range():
    if p_new_obs[i] not in (p_lower[i], p_upper[i]):
        anomalous_counter++;
if anomalous_counter > anomalous_threshold:
    observation_class = "Anomaly"
else
    observation_class = "Normal"
anomalous_counter = 0;
```

**FIGURE 3.** Pseudo-code routine of the decision policy.

- Accuracy ACC is the fraction of correctly identified results (attack and normal traffic)

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

- Precision PREC is the proportion of identified attacks that are indeed attacks

$$PREC = \frac{TP}{TP + FP}$$

### G. WORKFLOW SUMMARY

Figure 1 and Figure 2 schematically summarize our proposed workflow.

In particular, Figure 1 represents the workflow in the construction phase of the boundary polynomials, $p_{upper}(f)$ and $p_{lower}(f)$, necessary for the decision process to be applied in the "on-line" phase. In the "off-line" phase, the observations of the portion of the dataset used for the construction of the limit polynomials are processed to derive the scaling factors, $\min(\vec{f})$ and $\max(\vec{f})$, for each of the columns. Thereafter there is a coordinate transformation by application of PCA or MDS technique. The choice of PCA over MDS depends on the dataset, in particular, in our proposed flow the choice falls on the technique that reduces the size of the representation space the most. As a final step, we apply polynomial interpolation, from which we derive the coefficients needed to define the boundary polynomials to define the normal behaviour. The degree of the polynomial will be less or at most equal to the number of features downstream of PCA/MDS. The set of coefficients of the polynomials represent a new feature base. And being a non-linear application of the features derived from PCA/MDS, they also represent a further degree of uniqueness and hence of characterisation of normal versus abnormal behaviour.

As shown in Figure 2, the classification procedure ("on-line" because it acts on a single new observation) inherits the parameters for the characterisation of normal traffic, in order to apply a decision criterion based on the polynomial description of each new observation. In particular, the new observation is scaled with the values $\min(\vec{f})$, $\max(\vec{f})$ and subsequently transformed through the coordinate transformation matrices derived from the PCA/MDS reduction technique, from which an interpolating polynomial is derived to be finally compared with the boundary polynomials of the normal behaviour. The decision-making policy is based on calculating the points outside the previously calculated normality limits, comparing the polynomial associated with the new observation. Once a certain threshold, derived from preliminary statistical analysis, is exceeded, a decision is made to classify it as an anomaly.

## IV. EXPERIMENTAL RESULTS

This Section shows the results obtained by applying the method proposed in Section III to all 3 datasets described in Section II. The first step is applying the feature reduction techniques discussed in Section III.B

As shown in Fig.4, Fig.5 and Fig.6, the best result between PCA and MDS for the preliminary reduction of the problem size, depends on the dataset. In general, it is not possible to say a priori which one between PCA and MDS reduces the number of features more.

In Figure 4 it is shown that through the PCA technique 18 features have been obtained, in the new representation base. In particular we start from 42 features for the original KDD99, passing to about 70 features after applying the procedure of Encoding of the features to qualitative values, to then return, as shown, to 18 features to maintain at least 95% of the initial informative content.

In Figure 5 it is shown that through the PCA technique 20 features have been obtained, in the new representation base. In particular we start from 49 features for the original UNSW-NB15, passing to about 120 features after applying the procedure of Encoding of the features to qualitative values, to then return, as shown, to 20 features to maintain at least 95% of the initial informative content.

In Figure 6 it is shown that through the MDS technique 8 features have been obtained, in the new representation base. In particular we start from 80 features for the original CSE-CIC-IDS-2018, passing to about 150 features after applying the procedure of Encoding of the features to qualitative values, to then return, as shown, to 8 features to maintain at least 95% of the informative content.

In Figure 7 it is shown that through the PCA technique 14 features have been obtained, in the new representation base. In particular we start from 63 features for the original EDGE-IIOTSET 2022, passing to about 69 features after applying the procedure of Encoding of the features to qualitative values, to then return, as shown, to 14 features to maintain at least 95% of the initial informative content.

In Table 1 are summarized the number of new features obtained for each of the dataset in front of PCA and MDS technique application. To be noted that PCA achieves good performance in terms of feature reduction also for CSE-CIC-IDS-2018 and EDGE-IIOTSET 2022; hence PCA is a suitable technique to be adopted if the same feature reduction method must be used for all the different datasets.

Reasonably, the reduction in the number of features so marked is due to the fact that the original dataset is based on the characterisation of the flow of data and packets but also on the topology of the sub-net in which the data traffic circulates (i.e. on the numbers of input/output ports which are probably interpreted by the PCA/MDS as being of little informative significance).

The next step is to construct the polynomials containing the normal behaviour, which describe the upper and lower limits over the entire interpolation interval.

In the procedure to build the correct polynomial model for features interpolation are considered only the results obtained from the best preliminary feature reduction technique.

The quality of the result depends strongly on the degree of the chosen polynomial, which can be interpreted as a hyper-parameter of our method. This choice also depends on the dataset on which the classifier is applied.

Figure 8 shows the analysis of variation in the choice of the degree of the interpolating polynomial, in the case of the application of our method to the KDD99 dataset. The figure shows how the choice of this parameter is important for the efficiency in the process of discrimination between anomaly and normality of the analysed traffic. For example, in the particular case of using KDD99, the degree of the polynomial chosen to interpolate the values of the 18 features resulting from the PCA procedure is 10 (top right in the Figure 8).

Similarly, Figure 9 shows the graphical analysis of the different choice of the degree of the interpolating polynomial for the construction of the normality limits. In the particular case of UNSW-NB15, downstream of the PCA procedure there is a decrease of the problem size up to 20 features. Consequently, as can be seen in Figure 9, the best choice in terms of the degree of the interpolating polynomial is 10. Obviously the decision on the degree of the polynomials is made against the evaluation of the performance indices, in fact the graphical analysis serves for clarity of exposition to the reader.

As highlighted in Fig.8 and Fig.9, the quality of the interpolation depends on the choice of the degree of the polynomial. In fact, for too high degrees the typical Runge [41] phenomenon is revealed, also due to the choice of equally spaced interpolation nodes.

Quite analogous is the situation shown in Figure 10, where the interpolation result is compared when varying the degree of the polynomial for application to the CSE-CIC-IDS-2018 dataset. In this case, the feature reduction via MDS reaches up to problem size in the new representation base equal to 8, and the interpolating polynomial that returns the best result in terms of ROC performance indices, is 6.

In Figure 11 four polynomial interpolations about EDGE-IIOTSET 2022 are shown. As done with the previous three cases, the interpolations done are compared between

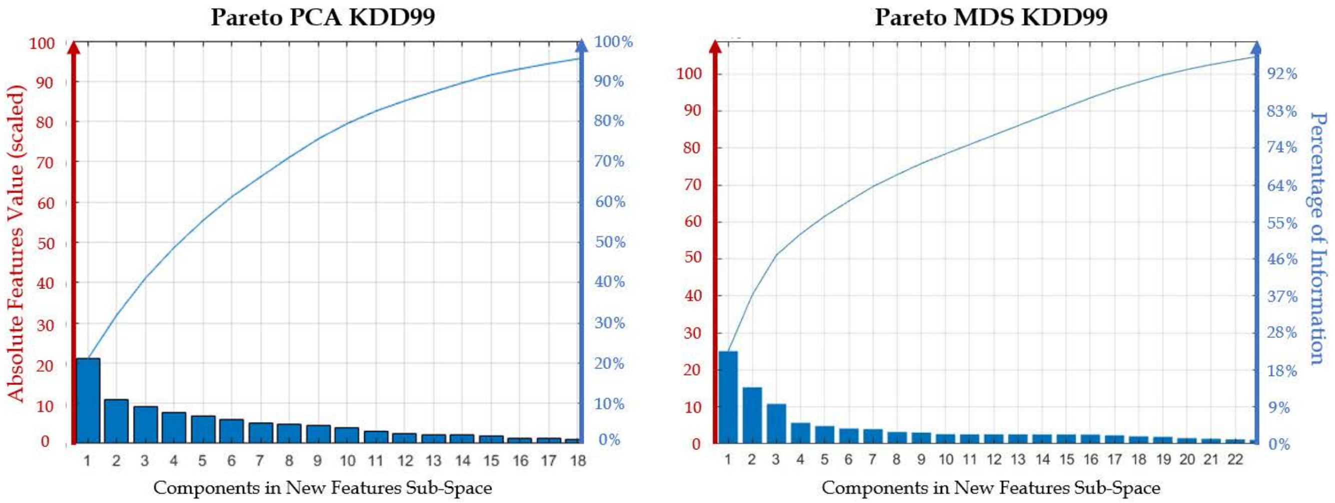

**FIGURE 4.** Comparison between PCA and MDS applied to KDD99, in preliminary features reduction phase.

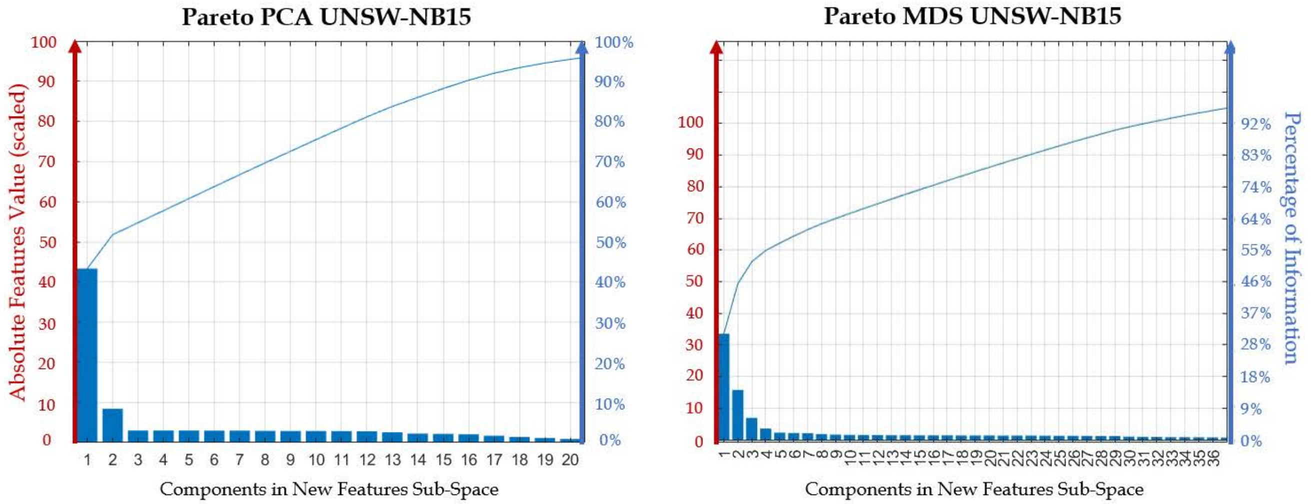

**FIGURE 5.** Comparison between PCA and MDS applied to UNSW-NB15, in preliminary features reduction phase.

**TABLE 1.** Comparison of the PCA and MDS techniques for features reduction.

| Technique | Size of the New Features Sub-Space | | | |
|---|---|---|---|---|
| | KDD99 | UNSW-NB15 | CSE-CIC-IDS-2018 | EDGE-IIOTSET 2022 |
| PCA | 18 | 20 | 10 | 15 |
| MDS | 22 | 36 | 8 | 42 |

them in order to determine the best polynomial degree that maximizes the ROC performance indices. In this case the best result is obtained with the polynomial degree equal to 8.

We would like to emphasise again that in the proposed workflow, the construction of the boundary polynomials for traffic normality is based only on data classified a priori as normal in the original datasets. In no way anomalous observations come into play in the process of constructing polynomials and decision thresholds.

Table 2 shows the results obtained by applying the technique proposed in this paper (Poly) vs ELM and SVM used as a one-class classifier. We denote ‘‘Poly BR’’ if the ‘‘best reduction’’ technique is chosen, and ‘‘Poly PCA’’ if PCA is chosen a priori. In Table 2 it is also reported the results obtained in case of non-optimal features reduction. In particular, for ‘‘Poly PCA’’ in case of CSE-CIC-IDS 2018 there is a bit of degradation in absolute performance but ‘‘Poly PCA’’ is still outperforming the state-of-art methods like ELM and SVM. This result suggests that it is possible to choose PCA a priori. In this way there is a less dependability from the dataset itself. Note also that the results in Table 2 are for the best choice among those shown, in terms of the degree of

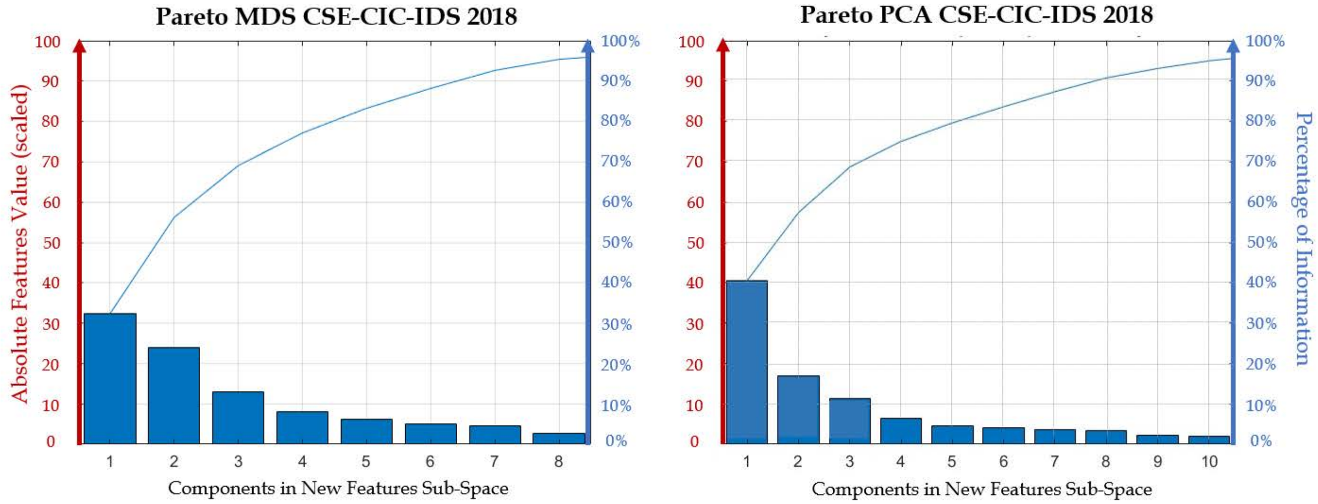

**FIGURE 6.** Comparison between PCA and MDS applied to CSE-CIC-IDS 2018, in preliminary features reduction phase.

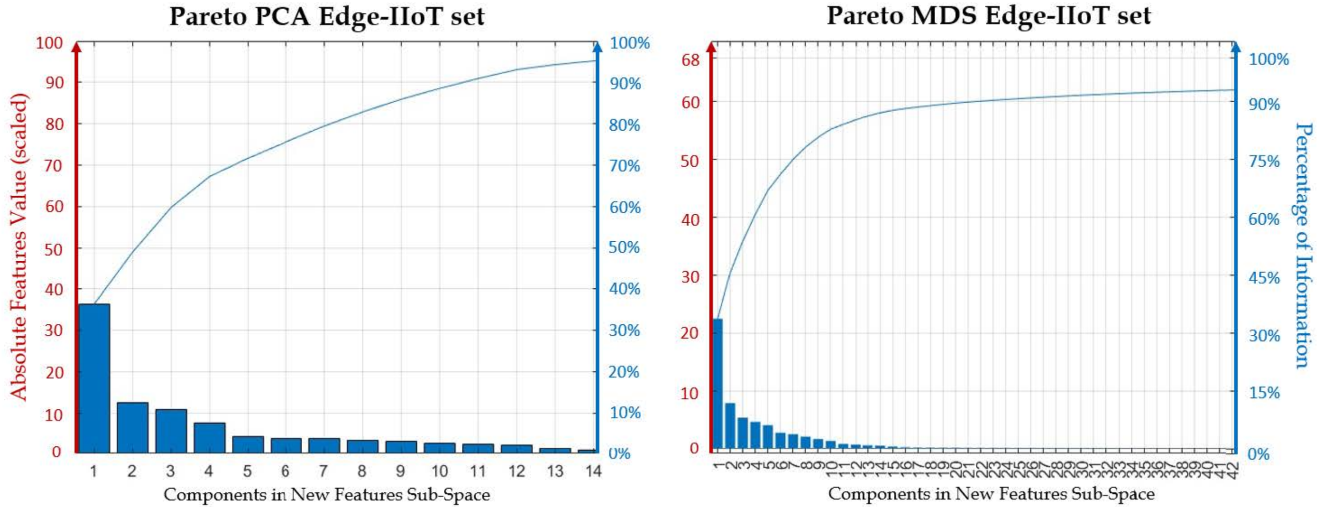

**FIGURE 7.** Comparison between PCA and MDS applied to EDGE-IIOTSET 2022, in preliminary features reduction phase.

the interpolating polynomial, and are for the ''off-line'' phase (i.e., the testing phase).

The comparison shows that the performance of our proposed method outperforms the other two models for all the considered datasets (KDD99, UNSW-NB15, CSE-CIC-IDS-2018 and EDGE-IIOTSET 2022) and for all the metrics defined in Section III.E. The most interesting result is the reduced false positive rate FPR compared to EML and SVM, which is in fact one of the crucial points in the development of new algorithms for anomaly detection.

It can also be stated that our results are fully comparable with the state of the art on ADS/IDS systems based on binary classifiers, which need to be trained with inputs from both classes (normal) and anomaly. In fact, [4]- [7] adopts KNN and ANN models and obtains an accuracy of 97% (only on KDD99 dataset); in [9]- [13] and [42], authors report the results of binary classifiers applied on UNSW-NB15 highlighting mean accuracy level around 95% with also some high FPR rate; in [14]- [16] and [43] it is reported a comparison of supervised machine learning (SVM, DT, DA) and deep learning (ANN, CNN, Autoencoder) models applied to CSE-CIC-IDS-2018, reveling an accuracy level close to 98%, for binary classifiers; in [35] the authors report the results of binary classifier applied on EDGE-IIOTSET 2022 using different type of machine learning (DT, RF, SVM) and deep learning (DNN) methods that provide an accuracy level of 99%. Summarizing, our method reaches a very similar level of detection performance, with low FPR respect some of literature results, with the advantage of no requests in terms of a priori knowledge on anomaly behaviour.

Note that, in order not have dependency issues with respect to the computational platform, the SVM and ELM models have been re-implemented following the design specifications reported by the authors cited in Section A.II.

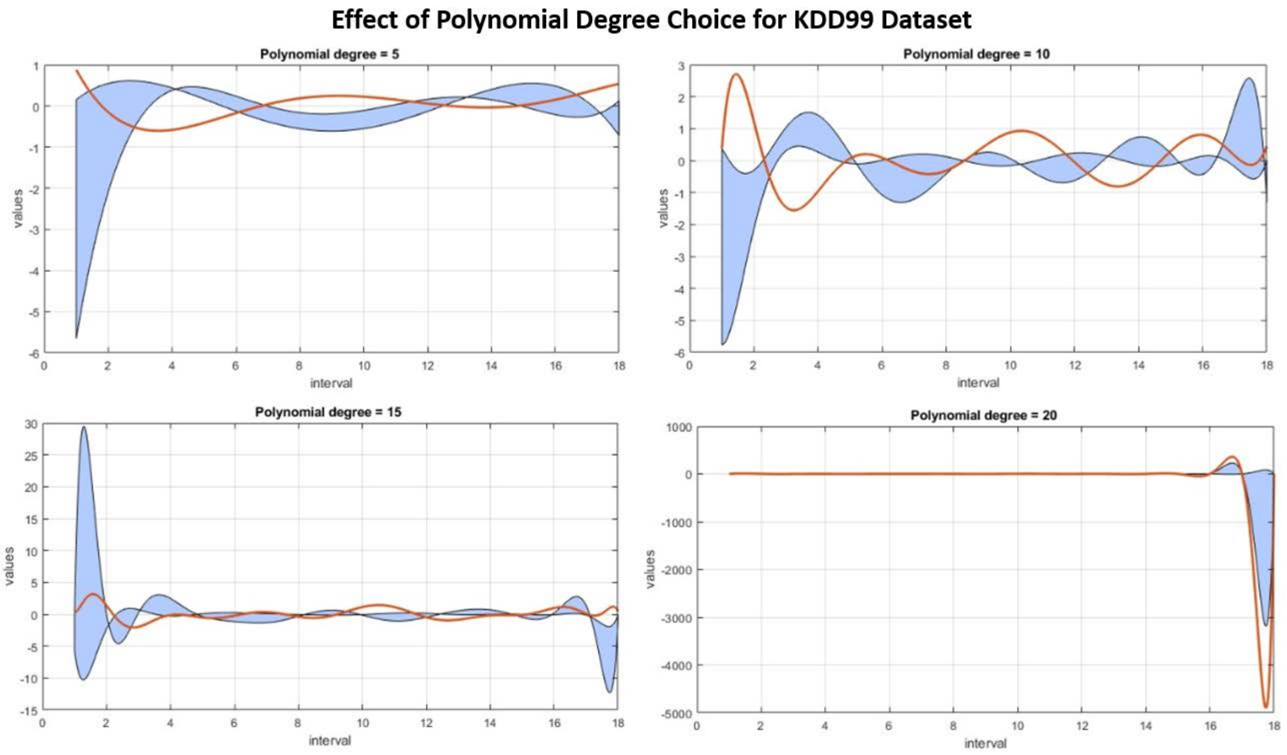

**FIGURE 8.** Comparison between Normal bound (blue) and the mean of the anomalous observations (red) within variation of the polynomial degree respect to KDD99 dataset.

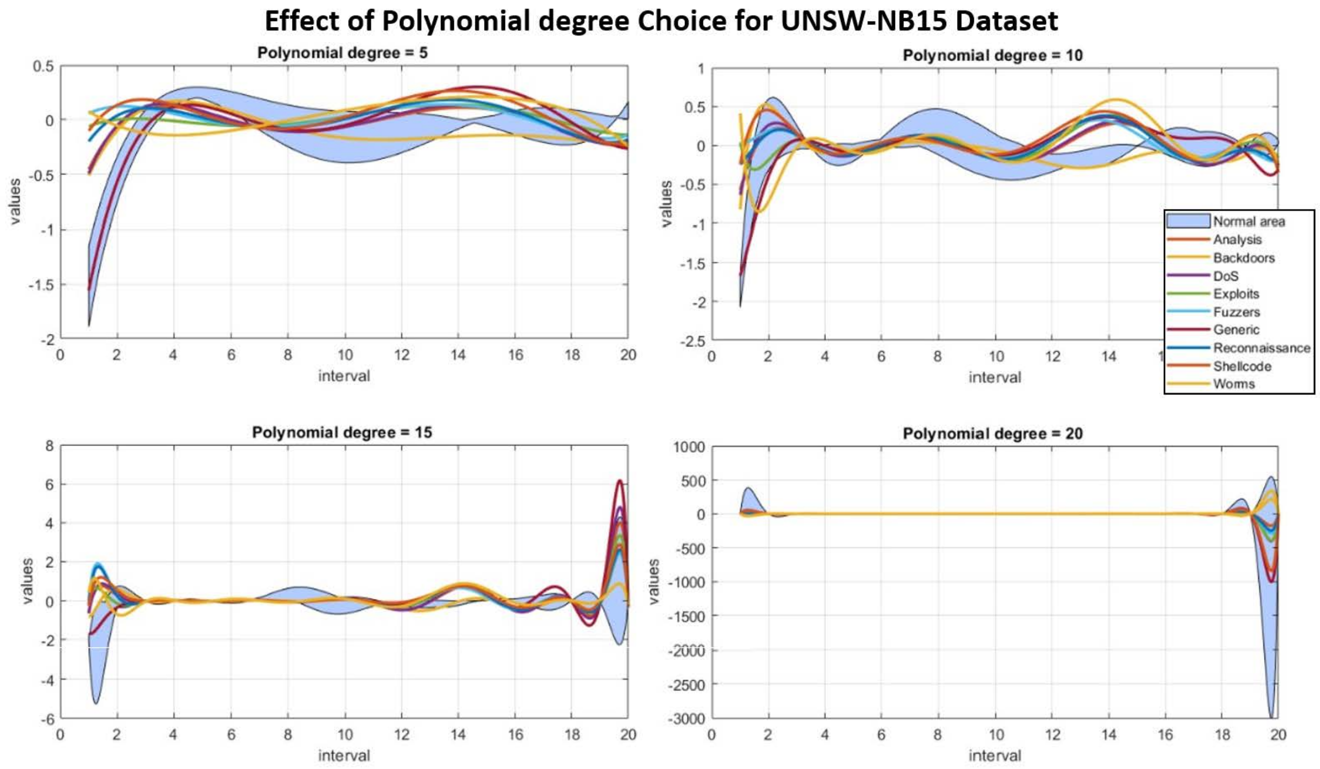

**FIGURE 9.** Comparison between Normal bound (blue) and the mean of the anomalous observations (see Legend) within variation of the polynomial degree respect to UNSW-NB15 dataset.

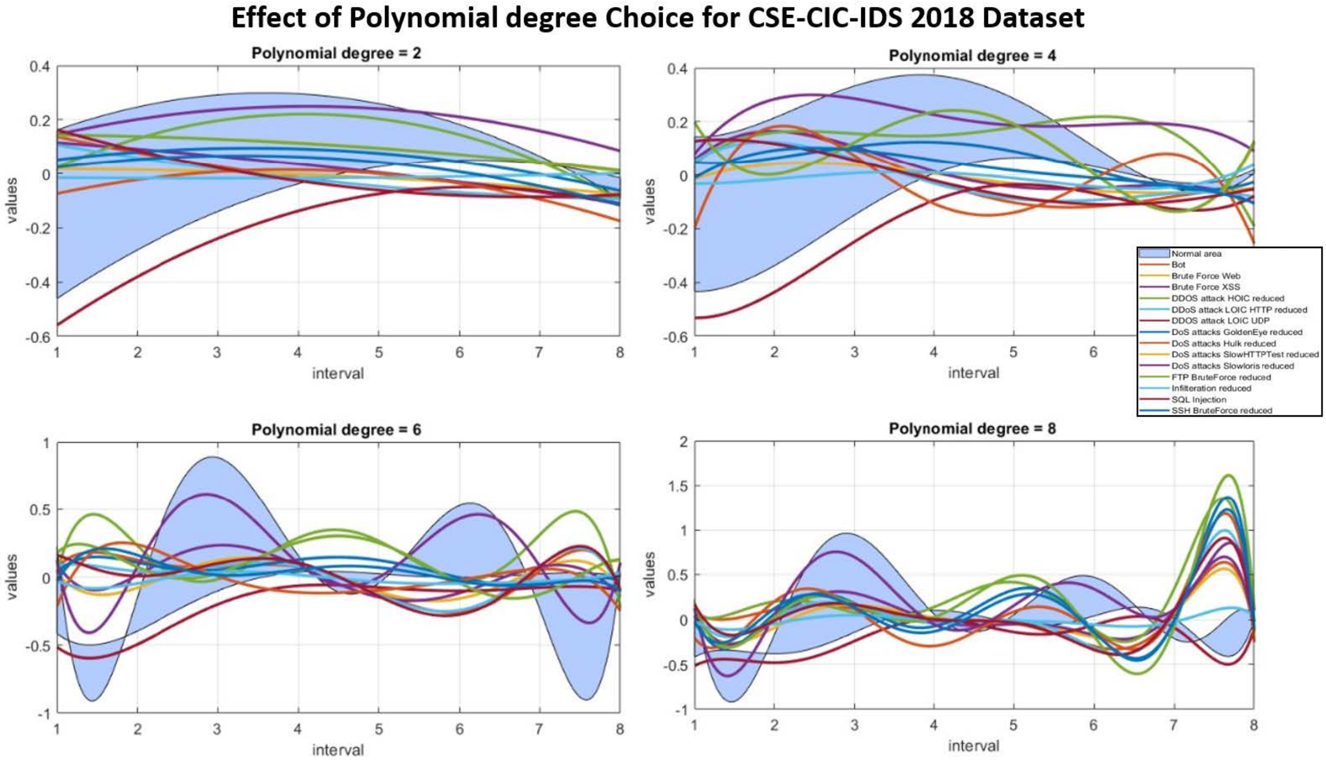

**FIGURE 10.** Comparison between Normal bound (blue) and the mean of the anomalous observations (see Legend) within variation of the polynomial degree respect to CSE-CIC-IDS 2018 dataset.

**TABLE 2.** Report of the obtained results and comparison vs the state of art.

| Model | DR% | | | | FPR% | | | | ACC% | | | | PREC% | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | KDD | UNSW | CIC | Edge-IIoT | KDD | UNSW | CIC | Edge-IIoT | KDD | UNSW | CIC | Edge-IIoT | KDD | UNSW | CIC | Edge-IIoT |
| SVM | 86.44 | 86.31 | 78.80 | 82.45 | 6.25 | 5.91 | 13.00 | 10.23 | 88.00 | 87.32 | 87.11 | 86.68 | 87.10 | 84.99 | 84.56 | 84.23 |
| ELM | 91.56 | 90.88 | 88.88 | 88.96 | 3.20 | 2.99 | 9.40 | 9.64 | 90.80 | 91.23 | 89.01 | 90.14 | 91.00 | 90.99 | 89.45 | 89.79 |
| Poly BR | 96.51 | 95.10 | 93.20 | 94.82 | 0.47 | 0.69 | 1.02 | 0.89 | 97.83 | 96.59 | 94.50 | 97.27 | 97.88 | 96.60 | 94.39 | 96.03 |
| Poly PCA | 96.51 | 95.10 | 92.11 | 94.82 | 0.47 | 0.69 | 1.76 | 0.89 | 97.83 | 96.59 | 92.54 | 97.27 | 97.88 | 96.60 | 92.33 | 96.03 |

The datasets were also processed through SNORT, configured through ‘‘community rules’’, obtaining much lower results in terms of accuracy. For KDD99 SNORT achieves about 61% accuracy, much lower than the 97.83% of our method; for UNSW-NB15 SNORT achieves about 39% vs. 96.59% of our method; for CSE-CIC-IDS SNORT achieves about 44% vs 95.5% of our method and for EDGE-IIOTSET 2022 achieves about 50% vs 97.27% accuracy level of our method. Note that SNORT can certainly be configured with custom rules to obtain better results. However, this highlights how rule-based tools are limited and lack flexibility for a user without specific knowledge of attack mechanisms. Summarizing, the method we propose is much more flexible than rule-based tools like SNORT or classic methods based on supervised-learning. Even in case of a new anomaly (never seen) the behavior will tend to go out of the confidence interval defined by $p_{upper}$ and $p_{lower}$. Therefore, even without knowing the specific mechanism of the new anomaly, it is possible to detect it.

A further analysis of the performance of the proposed method (Poly BR) is summarised in Table 3, in which it was studied how much time (average) is needed for the algorithm to process each new observation, intended as a feature vector.

The processing time of the proposed method is compared to those of the SVM and ELM one-class classifiers used as benchmarks. It should be noted that this processing time depends on the dataset, as operating times certainly depend on the numbers involved. The achieved results show that the processing time required by the proposed method (Poly BR) are lower than those for SVM and ELM classifiers for the UNSW-NB15, CSE-CID-IDS-2018 and EDGE-IIOTSET 2022. For the KDD99 dataset the processing time of the proposed method is comparable to the ELM technique and lower than the SVM classifier.

Computational times were also tested for a not optimal choice of the preliminary technique of feature reduction, highlighting that for KDD99, CSE-CIC-IDS-2018 and EDGE-IIOTSET 2022 the differences are not appreciable,

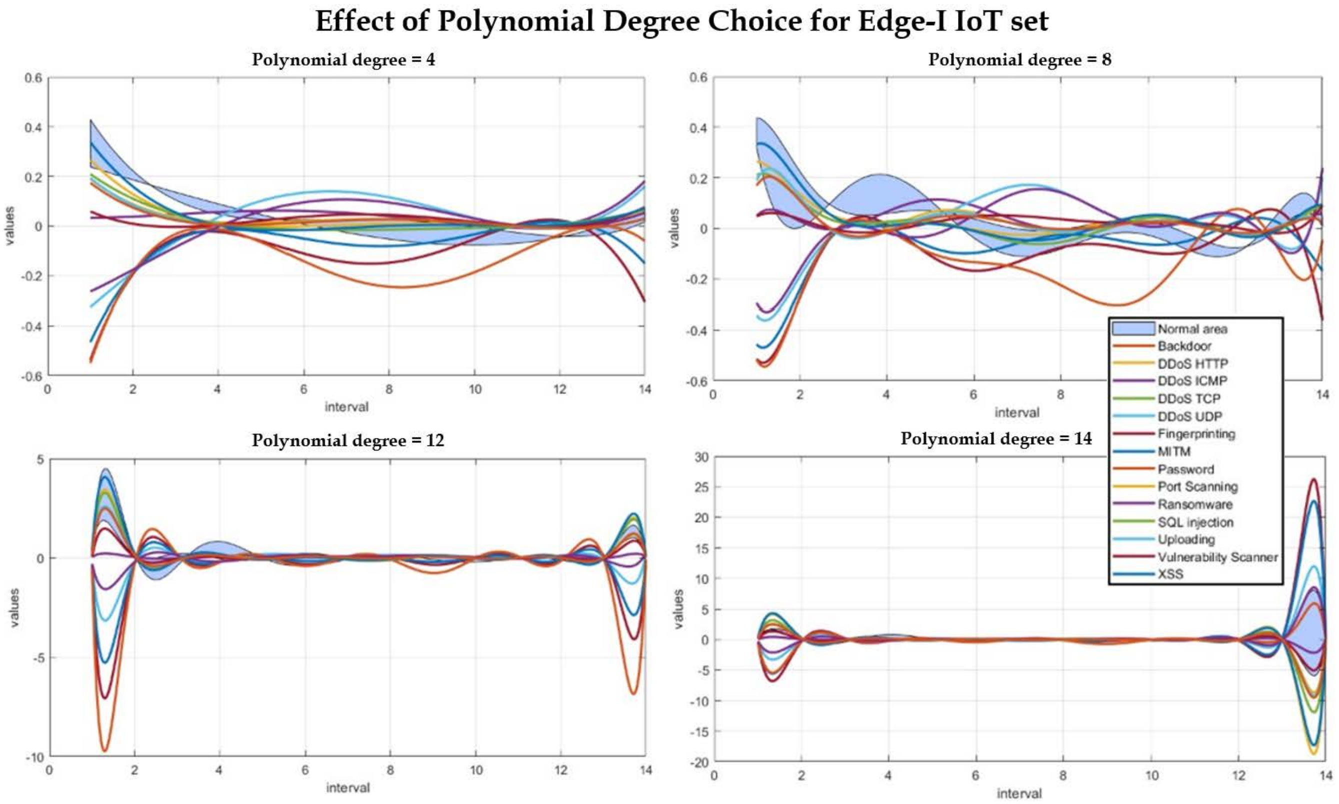

**FIGURE 11.** Comparison between Normal bound (blue) and the mean of the anomalous observations (see Legend) within variation of the polynomial degree respect to EDGE-IIOTSET 2022 dataset.

while there is an increase in processing times in case of UNSW-NB15, of about +10%. Reasonably, this is due to the difference in the number of features obtained downstream of the PCA and MDS techniques for the specific dataset. Obviously the difference in choice between PCA and MDS does not affect linearly the processing time, so the deterioration remains limited.

Notice that to verify that the processing time comparison between Poly, SVM and ELM is platform-independent, the test in Table 3 has been applied on two different processors, and we achieved the same results. The testing platforms were an Intel Core i3-6300 CPU 3.80 GHz with two cores (the one used to achieve the results in Table 2) and an Intel Core i7-8550U CPU 1.80 GHz with four cores.

## V. RESULTS DISCUSSION

As deeply discussed, the first step of the proposed method requires evaluating the Features Reduction techniques, in particular, we proposed PCA and MDS. Both techniques drastically reduce the number of features with respect original dataset and simplify the next phase of polynomial interpolation design. As shown in the previous section, the Features Reduction process strongly depends on the dataset, with widely different results. As discussed, the best choice is based on which one reduces the original dataset, maintaining the same information amount. Experimental results highlight that in KDD99 the two techniques have quite similar features in space reduction (18 with PCA vs. 22 with MDS); in UNSW-NB15 is highlighted a wide difference between the two techniques (20 with PCA vs. 36 with MDS); in CSE-CIC-IDS-2018, the two methods provides practically the same space reduction (10 with PCA vs. 8 with MDS); in EDGE-IIOTSET 2022 there is a wide difference between the two methods (15 with PCA vs. 42 with MDS). If for the user is necessary to define only one reduction technique, PCA results as the best choice for generalising the reduction phase. In this work, we propose polynomial interpolation as a further features selection & reduction method, applied after PCA or MDS. The idea of applying polynomial interpolation is to further empathize the differences between normal traffic behaviour and anomalies, introducing non-linear transformation. The polynomial interpolation phase requires an assessment procedure for the best choice of polynomials degree. Classification results are of course strongly dependent on the polynomial degree. The best choice is based on the most efficient combination of performance indexes (from ROC analysis). In particular, in KDD99 and UNSW-NB15 the degree of the optimal polynomial results is equal to 10 while in CSE-CIC IDS 2018 the optimal degree is 6 and in EDGE-IIOTSET 2022 the optimal degree is 8. In term of data interpretation, we can suppose that the complexity of the dataset increases as the number of features

**TABLE 3.** Computation time analysis.

| Dataset | Elapsed Time per Observation [$\mu s$] | | |
|---|---|---|---|
| | Poly BR | SVM | ELM |
| KDD99 | 223.8 | 288.8 | 210.4 |
| UNSW-NB15 | 229.3 | 479.5 | 287.7 |
| CSE-CIC-IDS 2018 | 480.2 | 869.8 | 795.6 |
| EDGE-IIOTSET 2022 | 466.5 | 782.0 | 790.4 |

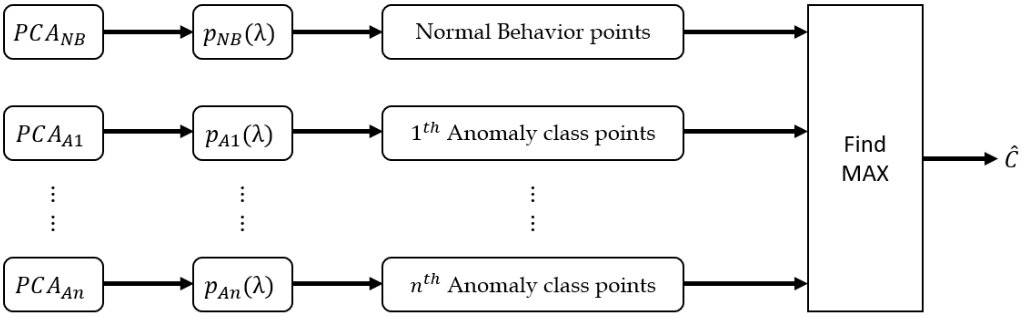

**FIGURE 12.** Proposed "Augmented" architecture for application of the proposed method in multi-class problems.

rises. From features reduction methods and degree of polynomial interpolation it follows that KDD99 (42 starting features and a polynomial degree of 10) and UNSW-NB15 (49 starting features and a polynomial degree of 10) have information complexity higher than CSE-CIC IDS 2018 (80 starting features and a polynomial degree of 8) and EDGE-IIOTSET 2022 (63 starting features and a polynomial degree of 8) even if they have an amount of features lower. This fact depends on the information content. Since each dataset was generated from different type of network traffic analyzer, the information content results a priory different. The data in CSE-CIC IDS 2018 are basically traffic statisticals and in EDGE-IIOTSET 2022 a big amount of records present are composed by zeros, meanwhile for KDD99 and UNSW-NB15 data are more related with nodes' interconnection inside of the same network and they have for each row a low numbers of zeros. From our point of view KDD99 and UNSW-NB15 result more complex than CSE-CIC IDS 2018 and EDGE-IIOTSET 2022 regarding to data interpretation. For further interpretation of the proposed workflow and results, several graphics on the features reduction and polynomial choice analysis are shown. The ROC indexes analysis reveals that the proposed method, based on an innovative one-class classifier, obtains higher performance concerning the SVM and ELM models. Notice that our method is based on statistical learning theory and numerical methods, which of course increase the interpretability of the entire workflow concerning fully AI-based methodology. Moreover, the best results in terms of computational time analysis suggest that the proposed method has a reduced computational effort than SVM and ELM, making it appropriate even for Embedded applications, unlike most of the work presented in the literature.

## VI. CONCLUSION AND FUTURE WORKS

In summary, this paper reports the procedure for the design of a one-class classifier, based on the concept of polynomial interpolation as a mathematical tool to insert uniqueness in anomaly recognition, which is not actually used in the literature. The entire workflow was presented, both from a formal and operational point of view, highlighting the advantages over the one-class classifiers used in the literature, such as ELM and SVM. We have shown the experimental results obtained by applying our proposed method on the four datasets. KDD99, UNSW-NB15 and CSE-CIC-IDS-2018 are most used in the literature for testing anomaly detection algorithms. The EDGE-IIOTSET 2022 is one of the newest dataset created by traffic extracted from a real IIoT network. We have shown that the algorithm we have developed achieves higher performance than the classical ELM and SVM that represent the standard for one-class classifiers. We also studied the computational complexity of the algorithm in terms of processing time for each observation, noting that even in this aspect, compared to SVM and ELM, the results are better. The proposed technique has been also compared for all considered datasets to rule-based IDS like SNORT and the achieved results show that our one-class classifier with polynomial interpolation leads to a much better accuracy. The presented work certainly leaves room for further extension and elaboration of the procedure. One of the goals is to define a version suitable also for multi-class problems in order to compare our workflow with ML/AI models based on supervised learning flow. Furthermore, the implementation on embedded platforms will be addressed in order to deal with safety problems also in applications of a different nature, such as in-vehicle communication systems and mechatronic systems of industrial interest.

As discussed in detail in the previous sections, the paper focuses on the design of a one-class classifier, as it is more important to detect anomalies rather than to specify their type. As attack scenarios are constantly evolving, this approach is crucial in safety-critical application contexts such as defense. In any case, as a future development of our proposed innovative method, it is certainly interesting to extend to the multi-class case. In particular, we propose the conceptual architecture shown in Figure 12.

Such architecture exploits the one-class method on several branches. In particular, assuming that the mechanisms of the anomalies are perfectly known and a sufficiently large dataset can be collected, it will be possible to extrapolate specific features that can be associated with the different predicted anomaly classes. Each branch of the architecture will thus handle membership in each specific class of the classification problem. The output of each of the branches will be the number of points within the confidence interval relative to the similarity with the polynomial associated with that class. This information is in fact quite equivalent to the output of a SOFTMAX layer in a neural network, which instead provides an estimate of the probability of membership in one of the classes of the problem. In the proposed architecture, the maximum among all values related to each branch will certainly be associated with the class that is closest in terms of polynomial representation.

## ACKNOWLEDGMENT

*(Pierpaolo Dini and Sergio Saponara contributed equally to this work.)*

## REFERENCES

[1] C. Rupa, G. Srivastava, S. Bhattacharya, P. Reddy, and T. R. Gadekallu, "A machine learning driven threat intelligence system for malicious URL detection," in *Proc. 16th Int. Conf. Availability, Rel. Secur.*, Aug. 2021, pp. 1–7.

[2] S. Gadamsetty, R. Ch, A. Ch, C. Iwendi, and T. R. Gadekallu, "Hash-based deep learning approach for remote sensing satellite imagery detection," *Water*, vol. 14, no. 5, p. 707, Feb. 2022.

[3] R. Ch, T. R. Gadekallu, M. H. Abidi, and A. Al-Ahmari, "Computational system to classify cyber crime offenses using machine learning," *Sustainability*, vol. 12, no. 10, p. 4087, May 2020.

[4] P. Dini and S. Saponara, "Analysis, design, and comparison of machine-learning techniques for networking intrusion detection," *Designs*, vol. 5, no. 1, p. 9, Feb. 2021.

[5] R. D. Ravipati and M. Abualkibash, "Intrusion detection system classification using different machine learning algorithms on KDD-99 and NSL-KDD datasets—A review paper," *Int. J. Comput. Sci. Inf. Technol.*, vol. 11, p. 16, Aug. 2019.

[6] P. Singh and A. Tiwari, "A review intrusion detection system using KDD'99 dataset," *Int. J. Eng. Res. Technol.*, vol. 3, no. 11, pp. 1103–1108, 2014.

[7] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, "Performance analysis of machine learning algorithms in intrusion detection system: A review," *Proc. Comput. Sci.*, vol. 171, pp. 1251–1260, Jan. 2020.

[8] P. Perera, P. Oza, and V. M. Patel, "One-class classification: A survey," 2021, *arXiv:2101.03064*.

[9] M. Belouch, S. El Hadaj, and M. Idhammad, "Performance evaluation of intrusion detection based on machine learning using apache spark," *Proc. Comput. Sci.*, vol. 127, pp. 1–6, Jan. 2018.

[10] A. Aleesa, M. Younis, A. A. Mohammed, and N. Sahar, "Deep-intrusion detection system with enhanced UNSW-NB15 dataset based on deep learning techniques," *J. Eng. Sci. Technol.*, vol. 16, no. 1, pp. 711–727, 2021.

[11] S. Moualla, K. Khorzom, and A. Jafar, "Improving the performance of machine learning-based network intrusion detection systems on the UNSW-NB15 dataset," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–13, Jun. 2021.

[12] A. Sonule, M. Kalla, A. Jain, and D. Chouhan, "UNSWNB15 dataset and machine learning based intrusion detection systems," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 3, pp. 2638–2648, 2020.

[13] S. M. Kasongo and Y. Sun, "Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset," *J. Big Data*, vol. 7, no. 1, pp. 1–20, Dec. 2020.

[14] M. P. Bharati and S. Tamane, "NIDS-network intrusion detection system based on deep and machine learning frameworks with CICIDS2018 using cloud computing," in *Proc. Int. Conf. Smart Innov. Design, Environ., Manage., Planning Comput. (ICSIDEMPC)*, Oct. 2020, pp. 27–30.

[15] M. A. Khan, "HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system," *Processes*, vol. 9, no. 5, p. 834, May 2021.

[16] P. Verma, A. Dumka, R. Singh, A. Ashok, A. Gehlot, P. K. Malik, G. S. Gaba, and M. Hedabou, "A novel intrusion detection approach using machine learning ensemble for IoT environments," *Appl. Sci.*, vol. 11, no. 21, Nov. 2021, Art. no. 10268.

[17] I. Ahmad, M. Basheri, M. J. Iqbal, and A. Raheem, "Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection," *IEEE Access*, vol. 6, pp. 33789–33795, 2018.

[18] J. Wang, S. Lu, S.-H. Wang, and Y.-D. Zhang, "A review on extreme learning machine," *Multimedia Tools Appl.*, pp. 1–50, May 2021, doi: 10.1007/s11042-021-11007-7.

[19] B. Lamrini, A. Gjini, S. Daudin, P. Pratmarty, F. Armando, and L. Travé-Massuyès, "Anomaly detection using similarity-based one-class SVM for network traffic characterization," in *Proc. 29th Int. Workshop Princ. Diagnosis*, 2018.

[20] K. Yang, S. Kpotufe, and N. Feamster, "An efficient one-class SVM for anomaly detection in the Internet of Things," 2021, *arXiv:2104.11146*.

[21] M. Verkerken, L. D'hooge, T. Wauters, B. Volckaert, and F. D. Turck, "Towards model generalization for intrusion detection: Unsupervised machine learning techniques," *J. Netw. Syst. Manage.*, vol. 30, no. 1, pp. 1–25, Jan. 2022.

[22] M. Sarhan, S. Layeghy, and M. Portmann, "Towards a standard feature set for network intrusion detection system datasets," *Mobile Netw. Appl.*, vol. 27, no. 1, pp. 357–370, Feb. 2022.

[23] S. Bagui, M. Walauskis, R. DeRush, H. Praviset, and S. Boucugnani, "Spark configurations to optimize decision tree classification on UNSW-NB15," *Big Data Cognit. Comput.*, vol. 6, no. 2, p. 38, Apr. 2022.

[24] A. Arqane, O. Boutkhoum, H. Boukhriss, and A. El Moutaouakkil, "A review of intrusion detection systems: Datasets and machine learning methods," in *Proc. 4th Int. Conf. Netw., Inf. Syst. Acad. Manage. Perspect. Secur.*, Apr. 2021, pp. 1–6.

[25] A. Thakkar and R. Lohiya, "A review of the advancement in intrusion detection datasets," *Proc. Comput. Sci.*, vol. 167, pp. 636–645, Jan. 2020.

[26] S. H. Kok, A. Abdullah, N. Z. Jhanjhi, and M. Supramaniam, "A review of intrusion detection system using machine learning approach," *Int. J. Eng. Res. Technol.*, vol. 12, no. 1, pp. 8–15, 2019.

[27] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1–22, Dec. 2019.

[28] A. Khraisat and A. Alazab, "A critical review of intrusion detection systems in the Internet of Things: Techniques, deployment strategy, validation strategy, attacks, public datasets and challenges," *Cybersecurity*, vol. 4, no. 1, pp. 1–27, Dec. 2021.

[29] G. Singh and N. Khare, "A survey of intrusion detection from the perspective of intrusion datasets and machine learning techniques," *Int. J. Comput. Appl.*, pp. 1–11, Feb. 2021, doi: 10.1080/1206212X.2021.1885150.

[30] M. S. Al-Daweri, K. A. Z. Ariffin, S. Abdullah, and M. F. E. Md. Senan, "An analysis of the KDD99 and UNSW-NB15 datasets for the intrusion detection system," *Symmetry*, vol. 12, no. 10, p. 1666, Oct. 2020.

[31] A. Özgür and H. Erdem, "A review of KDD99 dataset usage in intrusion detection and machine learning between 2010 and 2015," *PeerJ Preprints*, vol. 4, Jan. 2016, Art. no. e1954v1.

[32] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Nov. 2015, pp. 1–6.

[33] J. L. Leevy and T. M. Khoshgoftaar, "A survey and analysis of intrusion detection models based on CSE-CIC-IDS2018 big data," *J. Big Data*, vol. 7, no. 1, pp. 1–19, Dec. 2020.

[34] E. M. Campos, P. F. Saura, A. González-Vidal, J. L. Hernández-Ramos, J. B. Bernabé, G. Baldini, and A. Skarmeta, "Evaluating federated learning for intrusion detection in Internet of Things: Review and challenges," *Comput. Netw.*, vol. 203, Feb. 2022, Art. no. 108661.

[35] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40281–40306, 2022.

[36] M. J. Zaki and W. Meira Jr., *Data Mining and Machine Learning: Fundamental Concepts and Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2020.

[37] G. R. Naik, *Advances in Principal Component Analysis: Research and Development*. Cham, Switzerland: Springer, 2017.

[38] I. Borg, P. J. Groenen, and P. Mair, *Applied Multidimensional Scaling and Unfolding*. Cham, Switzerland: Springer, 2018.

[39] S. Chapra, *Applied Numerical Methods With MATLAB for Engineers and Scientists*. New York, NY, USA: McGraw-Hill, 2011.

[40] R. F. Mello and M. A. Ponti, *Machine Learning: A Practical Approach on the Statistical Learning Theory*. Cham, Switzerland: Springer, 2018.

[41] H. Lin and L. Sun, "Searching globally optimal parameter sequence for defeating Runge phenomenon by immunity genetic algorithm," *Appl. Math. Comput.*, vol. 264, pp. 85–98, Aug. 2015.

[42] D. Jing and H.-B. Chen, "SVM based network intrusion detection for the UNSW-NB15 dataset," in *Proc. IEEE 13th Int. Conf. ASIC (ASICON)*, Oct. 2019, pp. 1–4.

[43] S. Wankhede and D. Kshirsagar, "DoS attack detection using machine learning and neural network," in *Proc. 4th Int. Conf. Comput. Commun. Control Autom. (ICCUBEA)*, Aug. 2018, pp. 1–5.

**PIERPAOLO DINI** received the M.S. and Ph.D. (Hons.) degrees in automation engineering from the University of Pisa. He is currently a Postdoctoral Researcher at the Department of Information Engineering, University of Pisa. He collaborates in multiple European research projects focused on the development of advanced control and monitoring algorithms for mechatronic systems in industrial applications. His research interests include control systems technology, advanced theory of dynamic systems and control, advanced model-based design paradigms, optimal and nonlinear control, electrical machines and drives, and in-vehicle power electronics.

**ANDREA BEGNI** received the B.Sc. degree in management engineering and the M.Sc. degree in robotics and automation engineering from the University of Pisa, in 2015 and 2021, respectively. He is currently involved in research activities focused on machine learning and AI for the development of classification methods in the field of intrusion detection algorithms for computer and embedded systems with the University of Pisa.

**STEFANO CIAVARELLA** received the Ph.D. degree in computer science from the Sapienza University of Rome working on network and security topics. He currently works as a Cyber and Network Engineer with MBDA Italy. He is certified CCNA, CEH, and CSX. His research interests include networking, wireless sensor networks, network recovery, and network cyber security.

**EMILIANO DE PAOLI** is currently a Cyber Security Technical Expert at MBDA, Italy. In the Engineering Department of the Company, for many years he worked in national and international projects, especially in communications and cyber security domains. He is also leading the Research and Development National Team for cyber-security and communications areas, dealing with both the definitions of the technological road-maps and the financial aspects. Among his several duties, he personally follows collaborations with universities, research centers, and SMEs.

**GIUSEPPE FIORELLI** received the master's degree in electronic engineering from the University of Rome Tor Vergata, defending a thesis on software-defined radio. He is currently a Software Project Leader with MBDA Italy S.p.A., concerning with system integration and acceptance, and development of maintenance tools. He is leading a shift towards predictive maintenance, which builds on condition monitoring and prognostic capabilities, combining model-based and data-driven approaches to detect anomalies, isolate their cause and forecasting equipment remaining useful life. Moreover, he also introduced the concept of infrastructure as code in installation procedures, allowing for streamlined and repeatable equipment installation and identification of it's any deviation from nominal state. His work on monitoring, anomaly detection and system integrity brought him to cyber security, focusing on the identifications of signs of attacks and system compromise.

**CARMELO SILVESTRO** received the M.Sc. degree in electronic engineering from the University of Catania, Italy, in 2002. He is currently the Product Cyber Security Officer at MBDA, Italy. He took several professional certifications in cyber security (e.g. CISSP, CISA, CEH, CIFI, and L.A. ISO27001). Thanks to almost 20 years of professional experience, he has been supporting the technology roadmap definition on the cyber security domain, collaborating with research institutions and SMEs.

**SERGIO SAPONARA** (Senior Member, IEEE) received the master's *(cum laude)* and Ph.D. degrees from the University of Pisa. He is currently a Full Professor of electronics at the University of Pisa. He is the Director of the I-CAS Laboratory, Crosslab Industrial IoT, the Summer School Enabling Technologies for the IoT. He coauthored more than 300 scientific publications and 18 patents. He is the leader of many funded projects by EU and by companies like Intel, Magneti Marelli, Ericsson, and PPC. In 2012, he was a Marie Curie Research Fellow of IMEC. He is an IEEE Distinguished Lecturer and co-founder of special interest group on the IoT of both IEEE CAS and SP societies. He is an associate editor of several IEEE and Springer journals.