International Journal of Computational Intelligence Systems (2022) 15:50
https://doi.org/10.1007/s44196-022-00100-w

RESEARCH ARTICLE

# A Novel Hybrid Methodology for Anomaly Detection in Time Series

Lejla Begic Fazlic[1] · Ahmed Halawa[2] · Anke Schmeink[2] · Robert Lipp[2] · Lukas Martin[3] · Arne Peine[3] · Marlies Morgen[1] · Thomas Vollmer[4] · Stefan Winter[4] · Guido Dartmann[1]

Received: 2 February 2022 / Accepted: 24 June 2022

## Abstract

Numerous research methods have been developed to detect anomalies in the areas of security and risk analysis. In healthcare, there are numerous use cases where anomaly detection is relevant. For example, early detection of sepsis is one such use case. Early treatment of sepsis is cost effective and reduces the number of hospital days of patients in the ICU. There is no single procedure that is sufficient for sepsis diagnosis, and combinations of approaches are needed. Detecting anomalies in patient time series data could help speed the development of some decisions. However, our algorithm must be viewed as complementary to other approaches based on laboratory values and physician judgments. The focus of this work is to develop a hybrid method for detecting anomalies that occur, for example, in multidimensional medical signals, sensor signals, or other time series in business and nature. The novelty of our approach lies in the extension and combination of existing approaches: Statistics, Self Organizing Maps and Linear Discriminant Analysis in a unique and unprecedented way with the goal of identifying different types of anomalies in real-time measurement data and defining the point where the anomaly occurs. The proposed algorithm not only has the full potential to detect anomalies, but also to find real points where an anomaly starts.

**Keywords** Anomaly detection · Classification · Self Organizing Maps (SOM) · Linear Discriminant Analysis (LDA)

## Abbreviations

| | |
|---|---|
| SOM | Self Organizing Maps |
| LDA | Linear Discriminant Analysis |
| IoT | Internet of Things |
| ML | Machine learning |
| kNN | k-nearest neighbours |
| PSO | Particle swarm optimization |
| SVD | Singular value decomposition |
| PCA | Principal component analysis |
| LR | Logistic regression |
| DTW | Dynamic time warping |
| DTW-DBA | DTW Barycenter averaging |
| MIMICWF | MIMIC-III waveform database |
| MIMIC-III | Medical information mart for intensive care |

✉ Lejla Begic Fazlic
l.begic@umwelt-campus.de

Ahmed Halawa
hallawa@ice.rwth-aachen.de

Anke Schmeink
anke.schmeink@isek.rwth-aachen.de

Robert Lipp
robert.lipp@isek.rwth-aachen.de

Lukas Martin
lmartin@ukaachen.de

Arne Peine
apeine@ukaachen.de

Marlies Morgen
m.morgen@umwelt-campus.de

Thomas Vollmer
thomas.vollmer@philips.com

Stefan Winter
stefan.winter@philips.com

Guido Dartmann
g.dartmann@umwelt-campus.de

[1] Environmental Campus Birkenfeld, Trier University of Applied Sciences, Schneidershof, Trier 54293, Germany

[2] ISEK Research Area, RWTH Aachen University, Kopernikusstraße 16 52074, Aachen 52074, Germany

[3] Department of Intensive and Intermediate Care, University Hospital Aachen, Pauwelsstreet 30, Aachen 52074, Germany

[4] Philips GmbH Innovative Technologies, Pauwelsstreet 30, Aachen 52074, Germany

Published online: 22 July 2022

| | |
|---|---|
| SOFA | Sequential organ failure assessment |
| ICU | Intesive care unit |
| SKAB | Skoltech anomaly benchmark |
| NAB | Numenta anomaly benchmark |
| MQTT | Message queuing telemetry transport |

## 1 Introduction

With the increased use of sensor technology, a special focus of scientific research is anomaly detection. It is performed by defining a region within the value space that represents normal behaviour, and finding information that does not belong to those regions. Anomaly detection in the medical field is an important problem that has been extensively studied and various of different deep learning approaches has been proposed in different medical areas [1, 2]. Inconsistent behavior of the different anomaly is a major challenge e.g heart rate in particular context can be normal while in a different context could indicate a health concern. On the other side, dynamic changes in monitoring environments could yield to higher false positive rates where normal examples appear as abnormal. The ability to recognize the sepsis disease as one of the leading cause of death in the world condition, as soon as possible provides the best chances for recovery. Although the medical societies have proposed a new criteria for sepsis recognition [3], early detection and treatment of sepsis is still challenging. When sepsis is detected, the organ damage is already progressed and leads to potentially irreversible stage. The general symptoms and signs of sepsis are non-specific, so we should look for specific signs e.g anomalies that will point us to the etiology of sepsis. Automatically initiating timely medical interventions that include the use of antibiotics, intravenous fluids, and targeted recovery treatments can halve the risk of dying. Patients with suspected sepsis should be referred immediately to an appropriate medical facility for the treatment of sepsis. Early treatment of sepsis is cost-effective, reducing the hospital days of patients in intensive care units. There is no unique technique that can be sufficient to diagnose sepsis and the combinations of different approaches are needed. A different studies based on patient retrospective data [4] that includes more than 40 different patients features are used for optimal treatments strategy in sepsis disease and sepsis classification [5–7]. To the best of our knowledge, no study, based on the waveform vital sign parameters from patient bedside time series data, has been conducted in the context of sepsis disease and anomaly detection. However, our algorithm must be seen as complementary to other approaches based on laboratory signs and physician estimation. Early treatment of sepsis is cost-effective, reducing the hospital days of patients in intensive care units. We tried to detect the moments of abnormalities in patients times series data that could support physician decision and potentially leads to early treatment of sepsis disease. In our previous research, we developed a new hybrid algorithm for sepsis disease risk classification based on statistic, Dynamic Time Warping (DTW) and DTW Barycenter Averaging (DTW-DBA) [8]. Although we have obtained significant precision in risk detection within a certain interval, the disadvantage of our approach is the impossibility of accurately defining the beginning of the event. . Additionally, we also applied our algorithm for sensor signals based on integrated environmental sensors developed specifically for mobile applications and wearables. Since the Internet of Things (IoT) generates a tremendous amount of data, anomalies occur as part of such a system. They can be related to problems like indicators for critical situations in industrial systems or like detecting an abnormal behavior in medical devices or patient data. Therefore, it is very important both to recognize anomalies in an early stage and also to identify the time instance where anomalies starts.

In this paper, we present hybrid models that are able to detect anomalies and find the time instances when anomaly starts or stops. Our proposed algorithm integrates multiple scientific fields by combining statistical methods, SOM and LDA and can identify different types of anomalies in real-time measured data and defines the point at which the anomaly occurs. The results show that the proposed algorithm has not only the full potential to detect anomalies, but also to find real points when an anomaly starts.

## 2 Related Works

In real-time applications, we have usually an insufficient amount of abnormal observations for model training. On the other side, sometimes it is very difficult to generate anomalies according to the limitations of the system. Different statistical and Machine Learning (ML) methods are reported in the literature to address anomaly prediction issues, using historical and real-time data. The authors in [9] published a systematic review about different anomaly detection, analysis and prediction approaches based on ML and statistical methods. They collected different scientific methods used in an intelligent inhabitant environment, transportation systems, smart objects and healthcare systems. They critically appraise research studies and synthesize findings qualitatively and quantitatively. The authors in [10] suggested detection and data recovery based on a multivariate statistical analysis approach that exploits spatial density. Research study [11] uses a Gaussian mixture approach to detect the probabilistic abnormality in the sphere of motion, occupancy and door sensors. They use the mixture of a finite number of Gaussian distributions with unknown parameters to generate data points. Here the alerts are generated for specific activity event and weakness of this method is that no warning is generated in case of absence of activity event. The researchers in study [12, 13] proposed a hierarchical Markov model and a semi Markov model to predict sequential data and abnormal human behavior for embedded state sensors in smart home settings. An improved Hidden Markov model using frequent common patterns for anomaly detection is also suggested in medical studies [14, 15]. A major obstacle for using HMM for anomaly detection is the huge training time for behavioral model construction.

The authors [16] in recent published study proposed model that uses spurious correlation coefficient to calculate the graph entropy. The proposed model reduces the distance between two climate events with similar graph entropy. The focus of this research was to detect an abnormal graph from the dynamic graph instead of trying to detect changes. The proposed algorithm ignores the spatial information of the dynamic graph and could not recognize outliers with an abnormal neighbor structure. The study [17] proposed new approach for regression based on delegating classifiers to predict radon concentration in soil gas concentration and anomalies by delegating the samples to the next lower level that do not meet the desired threshold. The authors [18] employed SOM for data clustering in the first phase and then applied the shortest path algorithm to recognize anomalous events. A method based on statistical measure percentiles is used in supervised learning over the patterns of normal behavior to detect abnormal long periods of inactivity in a home [19]. Bayesian statistic and forecasting are used in many studies to define behavioral patterns, that are statistically estimated based on three probabilistic features: activation likelihood, sequence likelihood and event duration likelihood [20, 21]. As the authors conceded, that the amount of data to evaluate the models was small and should be increased.

Different ML techniques are also represented in the recent study to detect and predict anomaly behavior of the data. A new approach for anomaly detection using deep learning techniques with delayed prediction is represented in the recent study [22]. Authors in [23] introduced a novel computational approach based on Recurrence Quantification Analysis. This approach is suitable for multivariate time series data and provides multiple predicted value candidates and selects that closest to the measured value as the predicted value. Authors in [24] proposed a long short-term memory based anomaly detection method for discord search from univariate time series data. The structural features from normal training data are learned and then using statistical strategy based on the prediction error for observed data anomaly detection is performed. The main disadvantage of this approach is that it can not be directly address multivariate sequences and bias exists in the selection of public data.

Algorithms based on clustering techniques work by grouping similar objects in a cluster and then assuming that the anomalies do not belong to any cluster, or are very far from the central cluster, or belong to clusters with low gravity. Generally, k-nearest neighbours (kNN) anomaly detection schemes are classified into two categories: density-based and distance-based schemes [25]. Combinations of self-adaptive and dynamic k-means are also used for training data to learn weights prior to anomaly detection [26]. SOM are introduced for anomaly detection in combination with the kNN approach and Particle Swarm Optimization (PSO). The authors in [27] use the competitive learning process of the SOM and data-clustering technique in the anomaly detection. Singular Value Decomposition (SVD) is also one of the most popular methods for anomaly detection [28–30]. Algorithms based on principal component classifier - Principal Component Analysis (PCA) are also used with big-dimensional data [31, 32]. The hybrid model that combine Recurrent Neural Networks and Convolutional Neural Networks with SVD is presented in the recent study [33]. The power of the LDA and Logistic Regression (LR) is also used as approach to solve problems of detecting atypical objects [34, 35]. To our knowledge, our proposed hybrid combination of statistic, SOM and LDA has not yet been studied in context of time series data.

## 3 Methodology

### 3.1 Step 1: Data Sets

**Sensor data**: We used measurements from fifteen IoT kits containing Bosh BME680 environmental data sensors [36]. These sensors measure humidity, air quality and temperature. A Raspberry Pi was used as an access point and data sink for the IoT kits. The communication between the individual devices was performed via Message Queuing Telemetry Transport (MQTT). The Raspberry Pi publishes a measurement request via MQTT to what the kits listen and then respond with the measured values. The received measurements are then written into a database. All sensors were placed on a table in the middle of the 9 m$^2$ room.

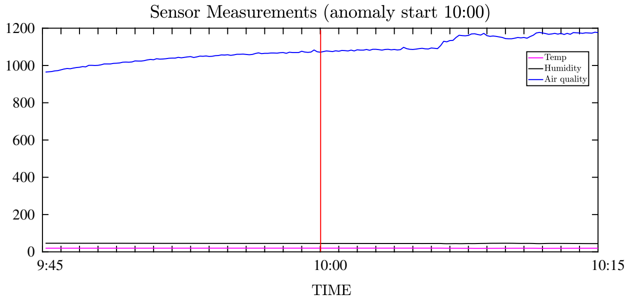


**Fig. 1** An example for signal representation- Sensor 1

The window and the opposite door were opened for 5 min each hour to create anomalous measurements. Afterwards they were closed again for 55 min, as it was found that this time was necessary for the readings to settle back to normal levels.This information is used to validate our model. We defined different data sets using 5 min intervals, 30 minutes intervals and randomly generated time intervals for testing and validation. All data measurements are stored in PostgresSQL database. An visualisation of sensors measurement is presented by Fig. 1 .

**Waveforms data**: As written in [4], the MIMIC-III Waveform Database (MIMICWF) contains contains a huge number of recordings of multiple physiologic signals and time series of vital signs collected in an Intesive Care Unit (ICU). Numeric data from MIMICWF typically include heart and respiration rate, SpO2, systolic, mean and diastolic blood pressure with others as available [4]. The three vital sign measurements are used in this study: heart rate, respiratory rate and mean arterial pressure. As in vital sign time series data, we could not get information about the status of the patients' diseases, we merged patients waveform data with Medical Information Mart for Intensive Care (MIMIC-III) numeric data to get already labeled data by the different Sepsis criteria [3]. Here we tried to recognize certain states of dysfunction and disease, but also tried to find the starting points of abnormal activity. The main idea behind our algorithm in the medical domain is to find out the best representative classes in the positive (disease) and negative (no-disease) direction.

The first decision to make for data splitting is to decide the proportion of data in the test set. Empirical studies show that the best results are obtained if we use 20–30% of the data for testing and the remaining 70–80% of the training [37, 38]. In our splitting choice, we followed the studies [39] that provided an explanation for this empirical result. Here, for fraction $p$ of the data that goes into the training set, the idea is to select a pone out of all possible values $p$, for which the product between $p$ and $(p-1)$ is the largest possible and it was for $p$ between 0.7 and 0.8. Here we considered two factors: sample size and computation intensity. The sample size is large enough, so we used 30% for test data and 70% for data training. In order to estimate how well our model has been trained and to estimate the model properties, we used an external validation data set.

### 3.2 Step 2: Data Transformation and SOM Classification

One of the most popular artificial neural network algorithm in the unsupervised learning category is the SOM. SOM is a type of the artificial neural network, whose training is performed by unsupervised learning in order to obtain a low-dimensional discrete representation of input patterns. This discrete representation of data is called a map. Self-organizing folders store information on the topological properties of inputs by using the function of adjacent neurons. This model was first described as a neural network by Teuvo Kohonen [40], therefore these networks are also called Kohonen maps. SOMs work in two phases: learning and mapping. Learning builds a map using input patterns. The mapping classifies the input vector. Our proposed hybrid model with steps described in detail is presented in Figs. 2, 3 and 4.

The first step in Fig. 2 represents data acquisition and the definition of data sets. First, we divided the data set into a positive data set (data set with anomaly) and a negative data set (data set without anomaly). Following study [39], we divide the data set in a training and in a testing part. We used time series measurements from $D$ different sensors $S_d$ with $n$ time series length. We mark with $N_f$ the total number of features used in our data set. Then, we can represent each feature's measurement, in the form Eq. (1):

$$\mathbf{f}_{i,d} = [f_{i,d}^{(1)}, ..., f_{i,d}^{(k)}, ..., f_{i,d}^{(n)}]^T \tag{1}$$

where $d$ denote sensor index, $i$ represent feature and $k$ is time index, respectively. Thus, each sensor measurement can be presented by Eq. (2)

$$S_d = [\mathbf{f}_{1,d}, \mathbf{f}_{2,d} .., \mathbf{f}_{N_f,d}]. \tag{2}$$

We provided descriptive statistics and an exploratory data analysis to find the approach to derive state space models from multiple time series data. Before we formed an input training vector for SOM, we represented data using the Hankel matrix [41]. Input-output data from Markov parameters are traditionally used to build the Hankel matrix, but there also exists the strategy where the Hankel matrix itself is

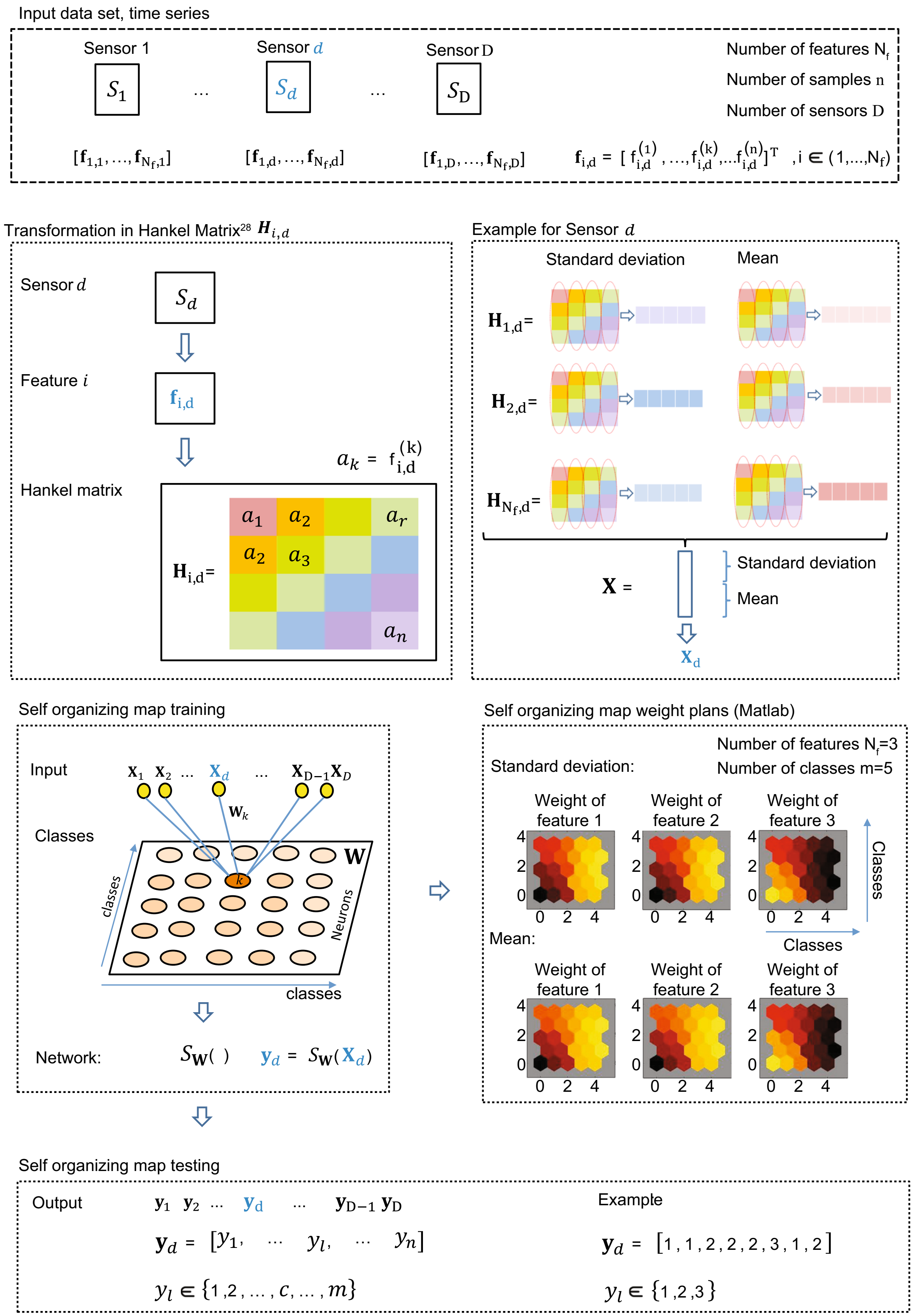

Input data set, time series
Sensor 1
Sensor $d$
Sensor D
$S_1$
…
$S_d$
…
$S_D$
Number of features $N_f$
Number of samples n
Number of sensors D
$[\mathbf{f}_{1,1},\ldots,\mathbf{f}_{N_f,1}]$
$[\mathbf{f}_{1,d},\ldots,\mathbf{f}_{N_f,d}]$
$[\mathbf{f}_{1,D},\ldots,\mathbf{f}_{N_f,D}]$
$\mathbf{f}_{i,d} = [\, f_{i,d}^{(1)}, \ldots, f_{i,d}^{(k)}, \ldots f_{i,d}^{(n)}]^T \;, i \in (1,\ldots,N_f)$
Transformation in Hankel Matrix28 $\boldsymbol{H}_{i,d}$
Sensor $d$
$S_d$
Feature $i$
$\mathbf{f}_{i,d}$
$a_k = f_{i,d}^{(k)}$
Hankel matrix
$\mathbf{H}_{i,d}=$
$a_1$
$a_2$
$a_r$
$a_2$
$a_3$
$a_n$
Example for Sensor $d$
Standard deviation
Mean
$\mathbf{H}_{1,d}=$
$\mathbf{H}_{2,d}=$
$\mathbf{H}_{N_f,d}=$
$\mathbf{X} =$
Standard deviation
Mean
$\mathbf{X}_d$
Self organizing map training
Input
$\mathbf{x}_1$ $\mathbf{x}_2$ … $\mathbf{x}_d$ … $\mathbf{x}_{D-1}\mathbf{x}_D$
$\mathbf{w}_k$
Classes
$k$
W
classes
Neurons
classes
Network:
$S_{\mathbf{W}}(\;)$
$\mathbf{y}_d = S_{\mathbf{W}}(\mathbf{X}_d)$
Self organizing map weight plans (Matlab)
Number of features $N_f$=3
Standard deviation:
Number of classes m=5
Weight of feature 1
Weight of feature 2
Weight of feature 3
4
2
0
0 2 4
Classes
Classes
Mean:
Weight of feature 1
Weight of feature 2
Weight of feature 3
4
2
0
0 2 4
Self organizing map testing
Output
$\mathbf{y}_1$ $\mathbf{y}_2$ … $\mathbf{y}_d$ … $\mathbf{y}_{D-1}$ $\mathbf{y}_D$
$\mathbf{y}_d = [y_1, \;\cdots\; y_l, \;\cdots\; y_n]$
$y_l \in \{1,2,\ldots,c,\ldots,m\}$
Example
$\mathbf{y}_d = [1,1,2,2,2,3,1,2]$
$y_l \in \{1,2,3\}$

**Fig. 2** (Algorithm design - 1.part )

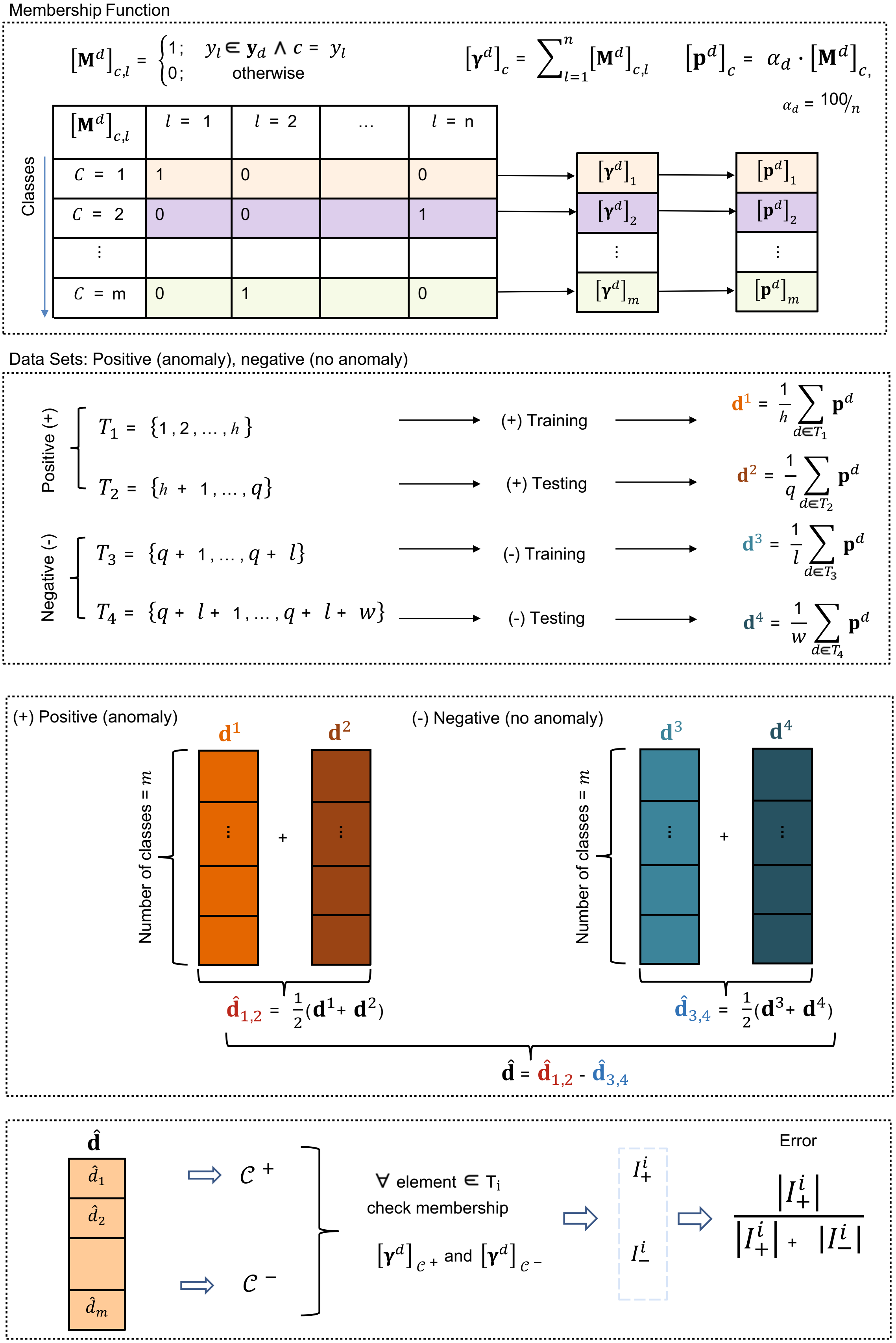

Membership Function
$[\mathbf{M}^d]_{c,l} = \begin{cases} 1; & y_l \in \mathbf{y}_d \wedge c = y_l \\ 0; & \text{otherwise} \end{cases}$
$[\boldsymbol{\gamma}^d]_c = \sum_{l=1}^{n} [\mathbf{M}^d]_{c,l}$
$[\mathbf{p}^d]_c = \alpha_d \cdot [\mathbf{M}^d]_{c,}$
$\alpha_d = 100/n$
$[\mathbf{M}^d]_{c,l}$
$l = 1$
$l = 2$
…
$l = n$
Classes
$C = 1$
$C = 2$
⋮
$C = m$
1
0
0
0
0
1
0
1
0
$[\boldsymbol{\gamma}^d]_1$
$[\boldsymbol{\gamma}^d]_2$
$[\boldsymbol{\gamma}^d]_m$
$[\mathbf{p}^d]_1$
$[\mathbf{p}^d]_2$
$[\mathbf{p}^d]_m$
Data Sets: Positive (anomaly), negative (no anomaly)
Positive (+)
$T_1 = \{1, 2, \ldots, h\}$
$T_2 = \{h + 1, \ldots, q\}$
Negative (-)
$T_3 = \{q + 1, \ldots, q + l\}$
$T_4 = \{q + l + 1, \ldots, q + l + w\}$
(+) Training
(+) Testing
(-) Training
(-) Testing
$\mathbf{d}^1 = \frac{1}{h} \sum_{d \in T_1} \mathbf{p}^d$
$\mathbf{d}^2 = \frac{1}{q} \sum_{d \in T_2} \mathbf{p}^d$
$\mathbf{d}^3 = \frac{1}{l} \sum_{d \in T_3} \mathbf{p}^d$
$\mathbf{d}^4 = \frac{1}{w} \sum_{d \in T_4} \mathbf{p}^d$
(+) Positive (anomaly)
(-) Negative (no anomaly)
$\mathbf{d}^1$
$\mathbf{d}^2$
$\mathbf{d}^3$
$\mathbf{d}^4$
Number of classes = $m$
Number of classes = $m$
+
+
$\hat{\mathbf{d}}_{1,2} = \frac{1}{2}(\mathbf{d}^1 + \mathbf{d}^2)$
$\hat{\mathbf{d}}_{3,4} = \frac{1}{2}(\mathbf{d}^3 + \mathbf{d}^4)$
$\hat{\mathbf{d}} = \hat{\mathbf{d}}_{1,2} - \hat{\mathbf{d}}_{3,4}$
$\hat{\mathbf{d}}$
$\hat{d}_1$
$\hat{d}_2$
$\hat{d}_m$
$\mathcal{C}^+$
$\mathcal{C}^-$
∀ element ∈ T$_i$
check membership
$[\boldsymbol{\gamma}^d]_{\mathcal{C}^+}$ and $[\boldsymbol{\gamma}^d]_{\mathcal{C}^-}$
$I_+^i$
$I_-^i$
Error
$\frac{|I_+^i|}{|I_+^i| + |I_-^i|}$

**Fig. 3** (Algorithm design - 2.part )

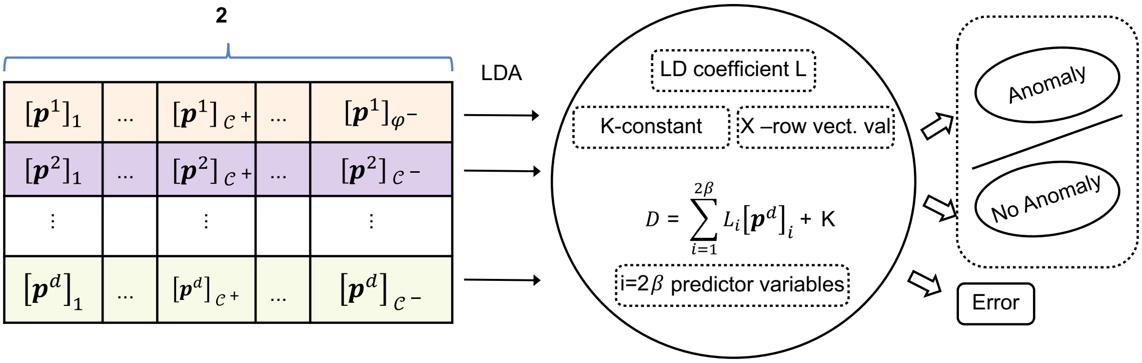

2
$[\boldsymbol{p}^1]_1$
...
$[\boldsymbol{p}^1]_{\mathcal{C}+}$
...
$[\boldsymbol{p}^1]_{\varphi-}$
$[\boldsymbol{p}^2]_1$
$[\boldsymbol{p}^2]_{\mathcal{C}+}$
$[\boldsymbol{p}^2]_{\mathcal{C}-}$
⋮
$[\boldsymbol{p}^d]_1$
$[\boldsymbol{p}^d]_{\mathcal{C}+}$
$[\boldsymbol{p}^d]_{\mathcal{C}-}$
LDA
LD coefficient L
K-constant
X –row vect. val
$D = \sum_{i=1}^{2\beta} L_i [\boldsymbol{p}^d]_i$ + K
i=2$\beta$ predictor variables
Anomaly
No Anomaly
Error

**Fig. 4** (Algorithm design - 3.part )

identified from input-output data as explained in [42]. For each sensor $S_d$ and each feature $i$, $i = (1, ..N_f)$, we constructed the Hankel matrix [41] $\mathbf{H}_{i,d}$, $i = (1, ..N_f)$ from $\mathbf{f}_{i,d}$, as it is represented in Eq. (3):

$$\mathbf{H}_{i,d} = \begin{bmatrix} a_1 & a_2 & \dots & a_r \\ a_2 & a_3 & .. & a_{r+1} \\ . & & & \\ . & & & \\ a_{n-r} & a_{n-r+1} & .. & a_n \end{bmatrix}, \text{where } a_k = f_{i,d}^{(k)} \tag{3}$$

where $a_n$ represents the last measurement in taken window size $n$. After we made transformation of input data in Hankel matrix trajectory, we calculated standard deviation $\sigma_d^i$ over the each column in Hankel matrix $\mathbf{H}_{i,d}$. Similarly, we calculated the mean value $\mu_d^i$ for each column in Hankel matrix $\mathbf{H}_{i,d}$. Using above described statistical transformation, the final form of each input data is represented by Eq. (4)

$$\mathbf{X}_D = \left[\sigma_{1,d}, \sigma_{2,d}, ..\sigma_{N_f,d}, \mu_{1,d}, \mu_{2,d}, ..\mu_{N_f,d}\right]^T \\ \text{where} \quad i = (1, ..N_f) \tag{4}$$

For each series separately, we repeated the same procedure. We used the statistic knowledge and the Hankel matrix and the transformed one dimensional time series in the multidimensional matrix Eq. (5)

$$\underline{X} = \left[\mathbf{X}_1, \mathbf{X}_2, .., \mathbf{X}_d, ..\mathbf{X}_D\right] \tag{5}$$

where $\underline{X}$ represents target input data for competitive learning (SOM). Here we trained SOM network $S_{\mathbf{w}}$ to compute the class vectors of each of the training inputs. According to [43], we used $M \sim 5\sqrt{N}$ number of classes, where $N$ represent number of observations and $M$ is the total number of neurons. Taking into account to keep data variability, we tried made more trials to define the best class size to reduce the maximum number of neurons. Using this approach, we could also take into account the various window size and different time series lengths.

## 3.3 Step 3: Evaluation of Best Classes

After applying data transformation and SOM unsupervised learning, our next steps, see Fig. 3, describe the approach to find the best representative classes for the state of anomaly and also the best representatives for the non-anomaly state. For each measurement $a_k$ of each input from the data set, using a trained SOM network, we defined a position as “1” if an element is a member of class$c$ and as “0” otherwise

We calculated the membership to each class as it is presented in pseudo-code:

**Algorithm 1** The part of proposed algorithm design

```
Initialize
$\mathbf{T_i}, i = (1, 2, 3, 4) \leftarrow$ data sets
$\mathbf{W_s} \leftarrow$ define window size length n
$\boldsymbol{H}_{i,d} \leftarrow$ define Hankel matrix
$\forall \mathbf{X}_d \leftarrow$ transform data and calculate std($\mathbf{X}_d$) and mean($\mathbf{X}_d$) in $\mathbf{H}_{i,d}$
Class $\leftarrow$ define number of SOM classes
$\mathbf{S_w} \leftarrow$train $\leftarrow$ configure SOM net parameters $\leftarrow \mathbf{X}_d \in (T_1 \cup \mathbf{T_3})$
begin
    $\forall \mathbf{X}_d \in T_i, i = (1, 2, 3, 4)$
    begin
        $\mathbf{y}_d = \mathbf{S_w}(\mathbf{X}_d) = [y_1, y_2, .., y_l, ..y_n]$, $y_l \in \{1, 2, .., c, ..m\}$
        $\forall y_l \in \mathbf{y}_d$ calculate the membership $\left[\mathbf{M^d}\right]_{c,l}$ to the class as :
        $\left[\mathbf{M^d}\right]_{c,l} = \begin{cases} 1, & y_l \in \mathbf{y}_d \wedge c = y_l, \\ 0, & otherwise \end{cases}$
        Calculate relative frequency distribution over the each class as
        $\left[\boldsymbol{\gamma}^d\right]_c = \sum_{l=1}^{n} \left[\mathbf{M}^d\right]_{c,l}$,
        $\left[\mathbf{p}^d\right]_c = \alpha_d \cdot \left[\mathbf{M}^d\right]_{c,l}$,where $\alpha_d = \frac{100}{n}$
    end
end
```

For each relative frequency distribution over the each class $\left[\mathbf{p}^d\right]_c$ we calculated the mean value over the each rows as it follows by Eq. (6):

$$\mathbf{d}^1 = \frac{1}{h}\sum_{d\in T_1}\mathbf{p}^d \quad \mathbf{d}^2 = \frac{1}{q}\sum_{d\in T_2}\mathbf{p}^d$$
$$\mathbf{d}^3 = \frac{1}{l}\sum_{d\in T_3}\mathbf{p}^d \quad \mathbf{d}^4 = \frac{1}{w}\sum_{d\in T_4}\mathbf{p}^d \tag{6}$$

After this step, four column vectors: $\mathbf{d}^1, \mathbf{d}^2$, $\mathbf{d}^3$ and $\mathbf{d}^4$ with the same dimension $(m \times 1)$ are obtained. Let's denote with $\hat{\mathbf{d}}_{1,2}$ the mean value between $\mathbf{d}^1$ and $\mathbf{d}^2$. Similarly, for $\hat{\mathbf{d}}_{3,4}$ we calculated the mean value between $\mathbf{d}^3$ and $\mathbf{d}^4$ as it follows Eq. (7):

$$\hat{\mathbf{d}}_{1,2} = \frac{1}{2}(\mathbf{d}^1 + \mathbf{d}^2)\hat{\mathbf{d}}_{3,4} = \frac{1}{2}(\mathbf{d}^3 + \mathbf{d}^4) \tag{7}$$

The calculated final difference in Eq. (8):

$$\hat{\mathbf{d}} = \hat{\mathbf{d}}_{1,2} - \hat{\mathbf{d}}_{3,4} = \left[\hat{d}_1, \hat{d}_2, ..., \hat{d}_m\right]^T \tag{8}$$

is used to find the best representative classes for anomaly and the best representative classes for non-anomaly. Let's define in Eq. (9) with $\mathcal{C}_+$ and $\mathcal{C}_-$ the best $\beta$ representative classes for anomaly and non-anomaly respectively:

$$\begin{aligned} \mathcal{C}_+ &= \{i \mid, \hat{d}_i \in D_{max}\}, D_{max} \subseteq \mathcal{D} \wedge \forall \hat{d}_i \in D_{max} : \hat{d}_i \geq \hat{d}_k, \\ &\qquad \mid D_{max} \mid= \beta, \hat{d}_k \in \mathcal{D} \setminus D_{max} \\ \mathcal{C}_- &= \{i \mid, \hat{d}_i \in D_{min}\}, D_{min} \subseteq \mathcal{D} \wedge \forall \hat{d}_i \in D_{min} : \hat{d}_i \leq \hat{d}_k, \\ &\qquad \mid D_{min} \mid= \beta, \hat{d}_k \in \mathcal{D} \setminus D_{min} \\ &\qquad \mathcal{D} = \{\hat{d}_1, \hat{d}_2, ..., \hat{d}_m\}\, 2\beta < m \end{aligned} \tag{9}$$

## 3.4 Step 4: Membership Analysis and Error Estimation

The next step is to check the cumulative membership to the best classes $\varphi_+$ and $\varphi_-$. Let's define by $[\boldsymbol{\gamma}^d]_{\mathcal{C}_+}$ and $[\boldsymbol{\gamma}^d]_{\mathcal{C}_-}$ the cumulative membership to the best classes in positive and negative direction defined by Eq. (10):

$$[\boldsymbol{\gamma}^d]_{\mathcal{C}_+} = \sum_{l=1}^{n}\sum_{c\,\in\,\varphi_+}\left[\mathbf{M^d}\right]_{c,l}[\boldsymbol{\gamma}^d]_{\mathcal{C}_-} = \sum_{l=1}^{n}\sum_{c\,\in\,\varphi_-}\left[\mathbf{M^d}\right]_{c,l} \tag{10}$$

For each training and testing element from the $T_i$, we calculated cumulative membership to the best classes in positive (anomaly) and negative (no-anomaly) direction. The final result will be as it follows by equations Eqs. (11) and (12):

$$\phi_1^+ = \begin{bmatrix} \gamma_{C_+}^{[1]} & \gamma_{C_-}^{[1]} \\ \gamma_{C_+}^{[2]} & \gamma_{C_-}^{[2]} \\ \dots & \\ \dots & \\ \gamma_{C_+}^{[h]} & \gamma_{C_-}^{[h]} \end{bmatrix} \phi_2^+ = \begin{bmatrix} \gamma_{C_+}^{[h+1]} & \gamma_{C_-}^{[h+1]} \\ \gamma_{C_+}^{[h+2]} & \gamma_{C_-}^{[h+2]} \\ \dots & \\ \dots & \\ \gamma_{C_+}^{[q]} & \gamma_{C_-}^{[q]} \end{bmatrix} \tag{11}$$

$$\phi_3^- = \begin{bmatrix} \gamma_{C_+}^{[q+1]} & \gamma_{C_-}^{[q+1]} \\ \gamma_{\mathscr{C}_+}^{[q+2]} & \gamma_{\mathscr{C}_-}^{[q+2]} \\ \dots & \\ \dots & \\ \gamma_{\mathscr{C}_+}^{[q+l]} & \gamma_{\mathscr{C}_-}^{[q+l]} \end{bmatrix} \phi_4^- = \begin{bmatrix} \gamma_{\mathscr{C}_+}^{[q+l+1]} & \gamma_{\mathscr{C}_-}^{[q+l+1]} \\ \gamma_{\mathscr{C}_+}^{[q+l+2]} & \gamma_{\mathscr{C}_-}^{[q+l+2]} \\ \dots & \\ \dots & \\ \gamma_{\mathscr{C}_+}^{[w]} & \gamma_{\mathscr{C}_-}^{[w]} \end{bmatrix} \tag{12}$$

Now, we calculate how many elements from (11) and (12) have $\gamma_{\mathscr{C}_+}^{[i]} > \gamma_{\mathscr{C}_-}^{[i]}$ and vice versa as it is defined in (13):

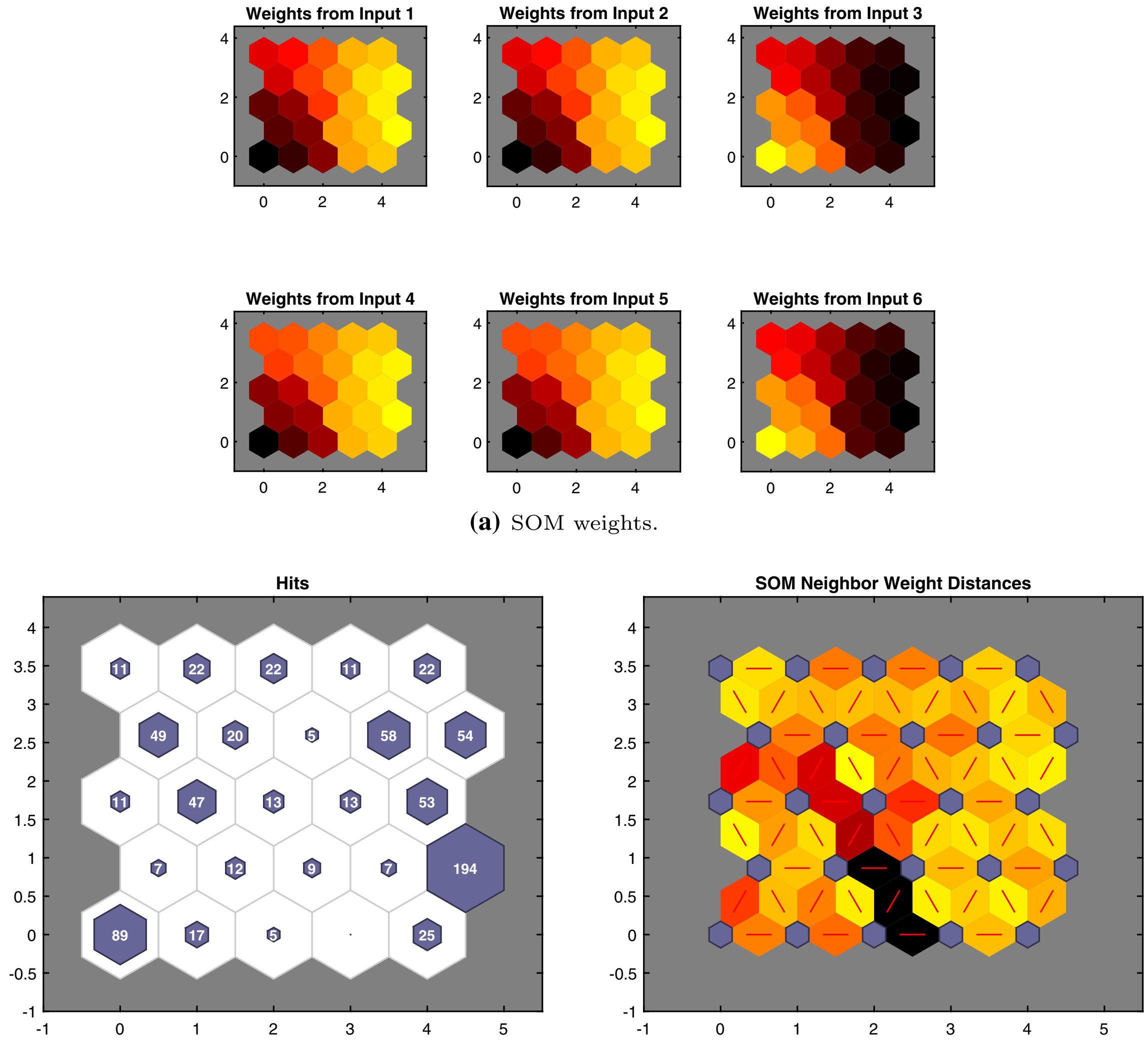
$$\begin{aligned} I_+^i &= \{d \ |, \gamma_{\mathscr{C}_+}^{[d]} > \gamma_{\mathscr{C}_-}^{[d]}, d \in T_i\} \\ I_-^i &= \{d \ |, \gamma_{\mathscr{C}_-}^{[d]} > \gamma_{\mathscr{C}_+}^{[d]}, d \in T_i\}; i = (1,2,3,4) \end{aligned} \tag{13}$$

For each dataset $T_i, i = (1,2,3,4)$ using (13), we calculated the cardinality of a set as it is presented in (14):

$$\sum_i^+ =| I_+^i | \sum_i^- =| I_-^i | \tag{14}$$

The final error is calculated by (15) as it follows:

$$E_i = \frac{| I_+^i |}{| I_+^i | + | I_-^i ||}, i = (1,2,3,4) \tag{15}$$



**(a)** SOM weights.

**(b)** SOM hits.

**(c)** SOM weights distance.

**Fig. 5** SOM visualization

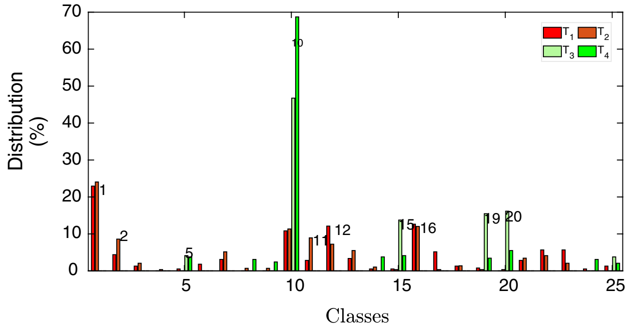


**Fig. 6** Distribution over the classes

### 3.5 Step 5: Linear Discriminant Model

After we finalized the previous steps, our next goal is to create a linear model representation using the LDA. The goal is to build a model as linear combination of the best predictors (in our case the best classes), that the best separates two classes (in our case: anomaly and no-anomaly). In this step we aim at investigating how to build a linear model from the previous steps using the best $\beta$ classes. LDA performs dimensional reduction while preserving as much of the class discriminatory information as possible. As we only have information that anomaly exists in a data set, without knowing the precise moment, we defined two classes 0 (no -anomaly) and 1 (anomaly). The training and testing data set for LDA contain the elements from $\left[\mathbf{p}^d\right]_c$ reduced to the best $\beta$ classes. Using the linear type of discriminant function, we classify [44, 45] each row of the data according to the relative frequency distribution over the best $\beta$ classes into one of the groups "+" or "−" and calculate the posterior matrix $\mathbf{P}$, unconditional predictive probability density of the sample observations $\mathbf{l}_{log}$ as well as constant $K$ and $\beta$ linear coefficients $L_i$ describing the boundary between the regions separating each pair of groups. The final model is presented in 4 and given by Eq. (16).

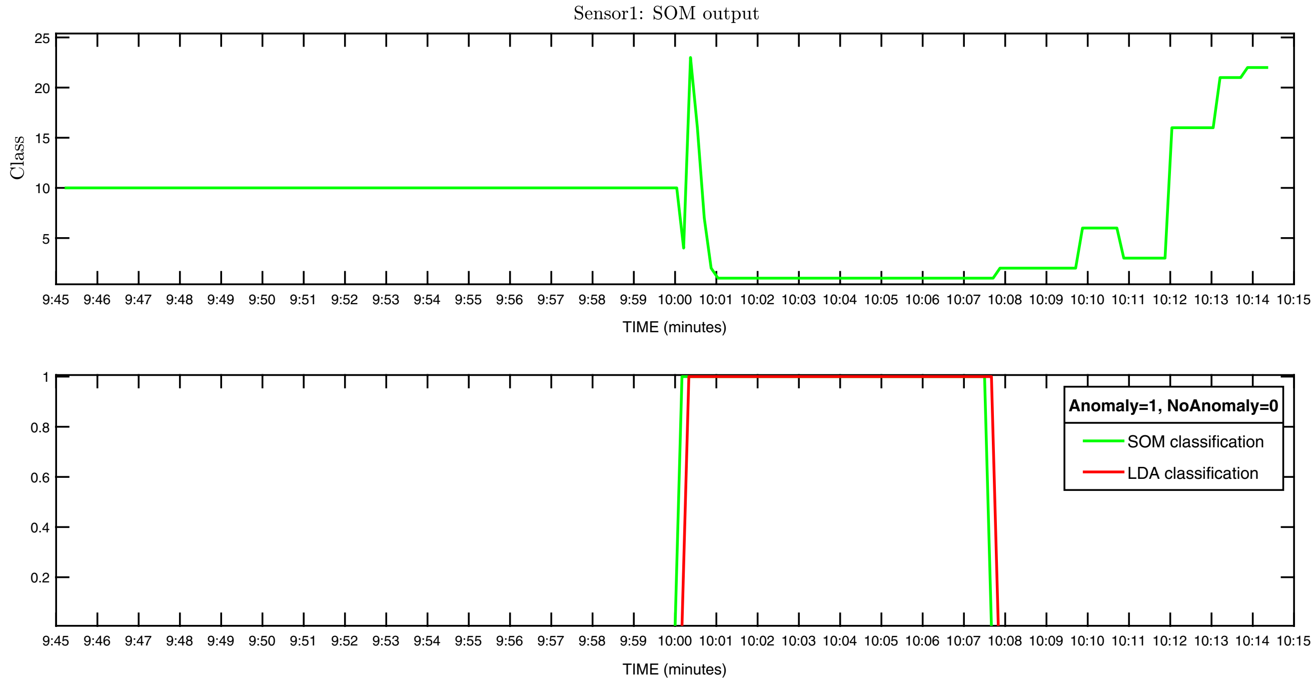


**Fig. 7** Anomaly detection for Sensor1 using proposed approach [40, 44]

**Table 1** The results: Angus criterion [3]

| Interval (h) | Positive | Negative | Precision | Recall | F1 | LDA error |
|---|---|---|---|---|---|---|
| 4 | 78 | 78 | 0.64 | 0.63 | 0.64 | 36.53 |
| 8 | 73 | 73 | 0.67 | 0.75 | 0.71 | 34.01 |
| 12 | 72 | 72 | 0.68 | 0.81 | 0.74 | 34.00 |
| 24 | 59 | 59 | 0.58 | 0.68 | 0.63 | 36.97 |

**Table 2** The results: Martin criterion [3]

| Interval (h) | Positive | Negative | Precision | Recall | F1 | LDA error |
|---|---|---|---|---|---|---|
| 4 | 33 | 33 | 0.68 | 0.79 | 0.73 | 29.10 |
| 8 | 33 | 33 | 0.77 | 0.73 | 0.75 | 27.61 |
| 12 | 33 | 33 | 0.74 | 0.78 | 0.76 | 27.69 |
| 24 | 25 | 25 | 0.73 | 0.88 | 0.80 | 27.00 |

**Table 3** The results: CDC criterion [3]

| Interval (h) | Positive | Negative | Precision | Recall | F1 | LDA error |
|---|---|---|---|---|---|---|
| 4 | 138 | 146 | 0.62 | 0.64 | 0.63 | 37.27 |
| 8 | 130 | 142 | 0.60 | 0.67 | 0.63 | 37.09 |
| 12 | 129 | 135 | 0.66 | 0.66 | 0.66 | 38.24 |
| 24 | 81 | 81 | 0.66 | 0.74 | 0.70 | 34.06 |

**Table 4** The results: Sepsis3 criterion [3]

| Interval (h) | Positive | Negative | Precision | Recall | F1 | LDA error |
|---|---|---|---|---|---|---|
| 4 | 210 | 73 | 0.82 | 0.69 | 0.75 | 35.74 % |
| 8 | 203 | 71 | 0.84 | 0.70 | 0.76 | 35.63 % |
| 12 | 192 | 67 | 0.80 | 0.68 | 0.74 | 36.25 % |
| 24 | 123 | 43 | 0.80 | 0.74 | 0.77 | 31.82% |

**Table 5** The results:The results: SOFA criterion [3]

| Interval (h) | Positive | Negative | Precision | Recall | F1 | LDA error |
|---|---|---|---|---|---|---|
| 4 | 240 | 43 | 0.92 | 0.79 | 0.85 | 32.85 |
| 8 | 231 | 42 | 0.92 | 0.78 | 0.84 | 31.41 |
| 12 | 224 | 41 | 0.91 | 0.77 | 0.83 | 31.53 |
| 24 | 135 | 26 | 0.91 | 0.72 | 0.80 | 28.81 |

**Table 6** The results: all criterion satisfied [3]

| Interval | Positive | Negative | Precision | Recall | F1 | LDA error |
|---|---|---|---|---|---|---|
| 90 min | 50 | 31 | 0.93 | 0.78 | 0.85 | 17.87 |
| 4 h | 49 | 31 | 0.91 | 0.86 | 0.88 | 17.40 |
| 8 h | 48 | 32 | 0.86 | 0.88 | 0.87 | 21.15 |
| 12 h | 44 | 29 | 0.89 | 0.89 | 0.89 | 19.23 |
| 24 h | 38 | 16 | 0.95 | 0.92 | 0.93 | 7.67 |

$$D = \sum_{i \in \{\mathscr{C}_+, \mathscr{C}_-\}} L_i \circ \left[\mathbf{p}^d\right]_i \tag{16}$$

where $\left[\mathbf{p}^d\right]_i$ represents relative frequency distribution for the best $\beta$ classes. Using the proposed approach, we used external validation data set to investigate how good the model could classify the new case in domain.

## 4 Results

In the first step, we implemented our approach using measurements collected over 15 sensors during a 24 h interval. To create different datasets, we defined datasets through different time intervals (5, 15 and 30 min). For example, we used 30 minutes data intervals with

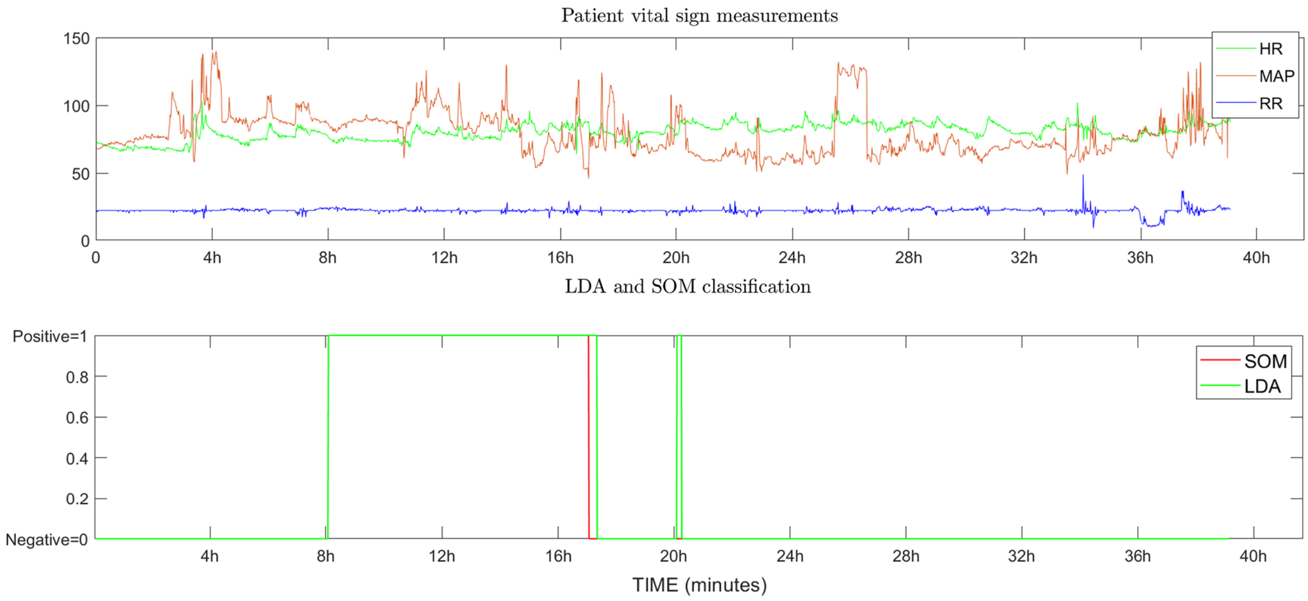


**Fig. 8** Vital sign measurements and statistic transformation

measurements collected over every 10 seconds from 15 sensors. Every hour, we generated 5 minutes of anomaly. The best results (presented in the next figures) are obtained using a 30 minutes data interval. We used as SOM parameter 25 classes and trained the SOM network. The SOM weights that presented the weight distribution from the *i*-th input to the layer's neurons are shown in Fig. 5a. The most negative connection is shown as black, zero connections as red, and the strongest positive connections as yellow. The neuron locations in the

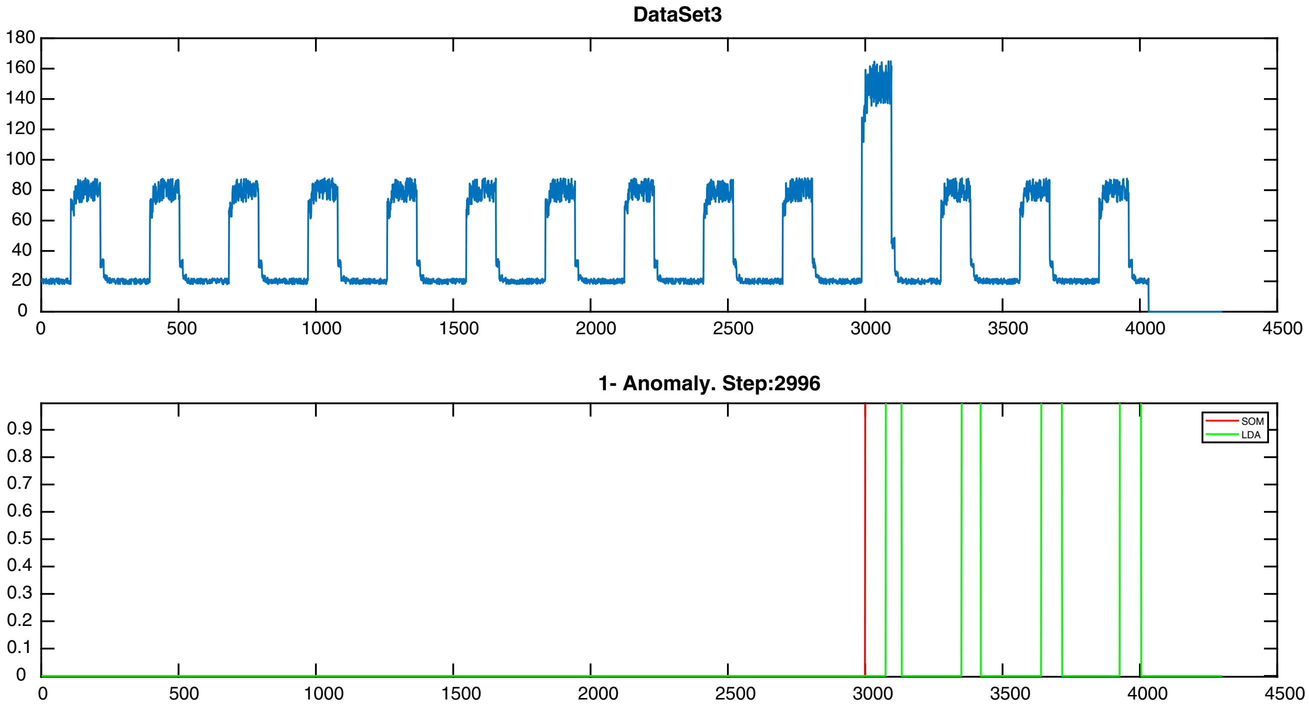


**Fig. 9** Anomaly detection (art_daily_jumpsup.csv) using proposed approach

**Table 7** The results of algorithm evaluation [46–48]

| Dataset | Precision | Recall | F1 | LDA error |
|---|---|---|---|---|
| SKAB [46] | 0.71 | 1 | 0.83 | 19.55 |
| YahooA1 [47] | 1 | 0.51 | 0.68 | 32 |
| YahooA2 [47] | 0.75 | 0.90 | 0.81 | 22 |
| YahooA3 [47] | 0.80 | 0.90 | 0.84 | 14.5 |
| NAB synthetic data [48] | 0.66 | 1 | 0.80 | 10 |

**Table 8** Average performance on the data set groups for different algorithms [50] vs our hybrid algorithm (H-algorithm)

| Dataset | Algorithm | Precision | Recall | F1 |
|---|---|---|---|---|
| YahooA1 | SVM | 0.46 | 0.76 | 0.58 |
| YahooA1 | kNN | 0.44 | 0.76 | 0.56 |
| YahooA1 | Arima | 0.41 | 0.75 | 0.73 |
| YahooA1 | AdVec | 0.44 | 0.48 | 0.46 |
| YahooA1 | **H-algorithm** | 1 | 0.51 | 0.68 |
| YahooA2 | SVM | 0.82 | 0.98 | 0.9 |
| YahooA2 | kNN | 0.48 | 0.98 | 0.65 |
| YahooA2 | Arima | 0.4 | 0.97 | 0.57 |
| YahooA2 | AdVec | 1 | 0.29 | 0.45 |
| YahooA2 | **H-algorithm** | 0.75 | 0.90 | 0.81 |
| YahooA3 | SVM | 0.83 | 0.64 | 0.72 |
| YahooA3 | kNN | 0.87 | 0.68 | 0.76 |
| YahooA3 | Arima | 0.66. | 0.54 | 0.60 |
| YahooA3 | AdVec | 1 | 0.02 | 0.03 |
| YahooA3 | **H-algorithm** | 0.80 | 0.90 | 0.84 |
| SKAB | Conv-AE [51] | | | 0.70 |
| SKAB | MSET [46] | – | – | 0.73 |
| SKAB | LSTM [46] | – | – | 0.64 |
| SKAB | **H-algorithm** | 0.71 | 1 | 0.83 |

topology that indicates how many of the training data are associated with each of the neurons are shown in Fig. 5b. The presented topology is a 5-by-5 grid, so there are in total 25 neurons. The maximum number of hits associated with any neuron is 194. Thus, there are 194 input vectors in that cluster. SOM Neighbor Weight Distances are described by Fig. 5c.The graphical representation of the best classes and histogram for the sensor example are given in Fig. 6. As result, we can conclude that the best five representative classes for anomaly are 1, 16, 12, 22 and 13, and the best five representative classes for non-anomaly are classes 10, 19, 5, 15 and 20.

As we have information about the existence of anomaly for intervals, we could validate the proposed approach not only to recognize anomaly, but also to find the time instance when the anomaly stars (time: 10:00, interval 9:45–10.15). Using LDA, we built a linear model using the best classes that we got in the previous steps. As we have information that each hour an anomaly starts and remains for the duration of 5 minutes, we can validate our approach and check how well our proposed model recognizes the anomaly and if it recognizes it, how precise it is with respect to the time point. The final output of the linear and SOM classification for the above mentioned time interval is given by Fig. 7.

We also tried to apply our algorithm over the patient's MIMICWF matched dataset. We applied our algorithm over 964 patients from matched subset of MIMICWF data. The 3 vital sign parameters are used in this study: heart rate, mean arterial pressure and respiratory rate. As the length of vital sign measurements is different for the individual patients, using our approach, we tested the model over different window size. We defined different window size (over 4, 8, 16 and 24 hours) and applied our algorithm over the data that are labeled by different criterion [3]: Angus, Martin, CDC, Sepsis3 and Sequential Organ Failure Assessment (SOFA). We also defined different numbers of classes to find the best representative classes for diseases.We tried to identify specific pattern for disease patients and using LDA we tried to identify interval points of "disease" behavior. Furthermore, we also created one data set where we selected patients that were positive (negative) for all criterion and then applied algorithm over that data set. The final results for patient dataset (the one admission) are represented in the Tables 1, 2, 3, 4, 5 and 6.

We tried to apply our algorithm for data set of 171 patients (Table 6), where patients are positive if they are labeled in all criterion as positive (102 patients) and similarly we found all patients that are labeled as negative (69 patients) by all criterion. Here we investigated also interval 100 min. The results are presented as follows:

The final results shows that the best sensitivity results by algorithm over 4 hours is for the SOFA criterion as it is represented in Table 5 . The worst results we get for the CDC criterion in Table 3. An example of original vital sign with the final results of classification is presented in Fig. 8.

Additionaly, we evaluate our anomaly detection algorithm using open sourced anomaly detection benchmarks: Skoltech Anomaly Benchmark (SKAB) [46], Yahoo Webscope [47] and Numenta Anomaly Benchmark (NAB)NAB [48]. SKAB dataset contain a multivariate time series collected from the sensors installed on the test bed. We used data from experiments with normal mode (no-anomaly) and data obtained from the other experiments (with anomaly). As it is explained [49], in NAB cases where univariate time series are included, the anomaly labeling mechanism is sometimes not relevant because window size is marked as anomalous even when in that window exists probably only 1–2 points that are anomalous. We used synthetic anomaly and no-anomaly NAB dataset. In the next Fig. 9, we present one result from anomaly labeled NAB data.

As the last anomaly benchmark dataset, we used public available Yahoo Webscope dataset, that contains synthetic as well as real data. The anomalies in Yahoo A1 Benchmark are marked by humans and therefore may not be consistent [47]. All other Yahoo dataset are sythetic. The highest F1 score is obtained for Yahoo A3 synthetic dataset and the lowest result is obtained using Yahoo A1 dataset. The results are presented in the Table 7:

We also investigated the other machine learning approaches and made a comparison to our hybrid approach (Table 8) . The results demonstrate that our algorithm shows comparable performance in Yahoo A1, Yahoo A3 and SKAB dataset with a slight favor for support vector machines for A2 dataset.

## 5 Conclusion

In our approach, we demonstrated how the hybrid model using statistical methods, neural networks and discriminant analysis could be applied to solve a complex anomaly problem not only in the area of smart sensors, but also in complex medical problems using medical big data. Our proposed model has potential to detect anomalies, positions of the objects and the real moment when the anomaly starts. Using the validation data set (time series data that are not presented to our model in the training or testing phase) the algorithm correctly identify anomalies and sensor types (number). The algorithm detects the real time point when anomaly starts with 93 percent of accuracy. In the case of only one sensor, the algorithm detects an existing anomaly 10 seconds later than it really starts. We also used time series data for vital signs and tried to apply our model to detect anomalies in patient behavior and to recognize diseases.

The algorithm shows also the potential to recognize “disease” behavior over the time series data where we use only 3 vital sign parameters. The best results of classification we got for patient data set where we included patients that were positive/negative by all criterion with precision of 0.95 and Recall over 0.92. Modeling and validation of a proposed approach is performed in Python, MATLAB and PostgreSQL environment. The next phase of our research will be focused on the fusion of the proposed model with genetic algorithms and Stochastic Petri Nets. Our goal is to optimize the parameters of the proposed model by genetic algorithms and apply it on the results derived by Stochastic Petri Nets to improve classification strategy.

**Author Contributions** LBF and GD identified the problem, conceived the methodology and designed the research. LBF evaluated the appropriate artificial methods to be employed in this research. LBF and AH performed the data extraction; AS, LM, AP and GD provided input on the methodology for analyzing the data; LBF and AH designed and implemented the simulation model. LBF carried out the mathematical analyses and provided the figures/tables; LBF, AH, AP, AS, RL, LM, MM, TV, SW and GD interpreted the data, analyzed and discussed results.

**Funding** Open Access funding enabled and organized by Projekt DEAL. This project was funded by Federal ministry of education and science (BMBF) IMEDALYTICS grant 13GW0280E, 13GW0280D, 13GW0280C, 13GW0280A and EIT Health- CLAIRE grant 19549. The funding institution of the study had no role in study design, data collection, data analysis, data interpretation or writing of the report.

**Data Availability** The MIMIC-III Waveform Database and MIMIC III are openly available. The databases were queried in the pgAdmin 4 version. The pre-processing and all computations are done in Python 3.7 and in Matlab version 2019a. Access to the sensor data used in our research is available via https://gitlab.rlp.net/l.begic899724/anomalydetection.

## Declarations

**Conflict of Interest** A.P. and L.M. are co-founders and chief executive officers of Clinomic GmbH. AS and GD are former co-founder of Clinomic and are now Scientific Advisors of Clinomic GmbH. All remaining authors declare that they have no conflict of interests.

**Ethical Approval** Not applicable.

**Consent to Participate** Not applicable.

**Consent for publication** Not applicable.

## References

1. Chalapathy, R., Chawla, S.: Deep learning for anomaly detection: a survey (2019). https://doi.org/10.48550/arXiv.1901.03407
2. Thudumu, S., Branch, P., Jin, J., Singh, J.J.: A comprehensive survey of anomaly detection techniques for high dimensional big data. J. Big Data **7**, 1–30 (2020)

3. Alistair, E.: A comparative analysis of sepsis identification methods in an electronic database. Crit. Care Med. **2018**(46), 494–499 (2018). https://doi.org/10.1097/CCM.0000000000002965
4. Goldberger, A.: Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. Circulation **101**, 215–220 (2000). https://doi.org/10.1161/01.cir.101.23.e215
5. Krissaane, I., Hampton, K., Alshenaifi, J., Wilkinson, R.: Anomaly detection semi-supervised framework for sepsis treatment. In: 2019 Computing in Cardiology (CinC), 1–4 (2019). https://doi.org/10.23919/CinC49843.2019.9005527
6. Komorowski, M., Celi, L.A., Badawi, O.: The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. Nat. Med. **24**, 1716–1720 (2018). https://doi.org/10.1038/s41591-018-0213-5
7. Hou, N., Li, M., He, L.: Predicting 30-days mortality for mimic-iii patients with sepsis-3: a machine learning approach using xgboostthe artificial. J. Transl. Med. **18**, 462 (2020). https://doi.org/10.1186/s12967-020-02620-5
8. Begic Fazlic, L.e.a.: A machine learning approach for the classification of disease risks in time series. In: 9th Mediterranean Conference on Embedded Computing (MECO) (2020). https://doi.org/10.1109/MECO49872.2020.9134275
9. Fahim, M., Sillitti, A.: Anomaly detection, analysis and prediction techniques in iot environment: a systematic literature review. IEEE Access **7**, 81664–81681 (2019). https://doi.org/10.1109/ACCESS.2019.2921912
10. Magán-Carrión, R., Camacho, J., García-Teodoro, P.: Multivariate statistical approach for anomaly detection and lost data recovery in wireless sensor networks. Int. J. Distrib. Sens. Netw. **7**, 1–20 (2015). https://doi.org/10.1155/2015/672124
11. Cardinaux, F., Brownsell, S., Hawley, M., Bradley, D.: Modelling of behavioural patterns for abnormality detection in the context of lifestyle reassurance. CIARP: Iberoamerican Congress Pattern Recognition, 243–251 (2008). https://doi.org/10.1007/978-3-540-85920-8_30
12. Duong, T.V.., Bui, H.H., Phung, D.Q., Venkatesh, S.: Activity recognition and abnormality detection with the switching hidden semi-markov model. Proc. IEEE Comput: Soc. Conf: Comput. Vis. Pattern Recognition CVPR 7, 838–845 (2005). https://doi.org/10.1109/CVPR.2005.61
13. Kang, D. W.and Shin, Do., S.: Detecting and predicting of abnormal behavior using hierarchical markov model in smart home network. 2010 IEEE 17Th International Conference on Industrial Engineering and Engineering Management, 410–414 (2010). https://doi.org/10.1109/ICIEEM.2010.5646583
14. Sultana, A., Abdelwahab, H.L., Couture, M.: An improved hidden markov model for anomaly detection using frequent common patterns. IEEE International Conference on Communications (ICC), 1113–1117 (2012). https://doi.org/10.1109/ICC.2012.6364527
15. Zhu, Y.: Automatic detection of anomalies in blood glucose using a machine learning approach. J. Commun. Netw. **13**, 92–97 (2011). https://doi.org/10.1109/IRI.2010.5558959
16. Li, G., Jung, J.J.: Entropy-based dynamic graph embedding for anomaly detection on multiple climate time series. Sci. Rep. **11**, 13819 (2021). https://doi.org/10.1038/s41598-021-92973-8
17. Rafique, M., Tareen, A.D.K., Mir, A.: Delegated regressor, a robust approach for automated anomaly detection in the soil radon time series data. Sci. Rep. (2020). https://doi.org/10.1038/s41598-020-59881-9
18. Kim, J., Cho, J.: An online graph-based anomalous change detection strategy for unsupervised video surveillance. EURASIP J. Video Process. (2019). https://doi.org/10.1186/s13640-019-0478-8
19. Cuddihy, P., Weisenberg, J., Graichen, C.M., Ganesh, M.: Algorithm to automatically detect abnormally long periods of inactivity in a home. Proceedings of the 1st ACM SIGMOBILE International Workshop on Systems and Networking Support for Healthcare and Assisted Living Environments, 89–94 (2007). https://doi.org/10.1145/1248054.1248081
20. Ordóñez, F.J., Toledo, P., Sanchis, A.: Sensor based Bayesian detection of anomalous living patterns in a home setting. Pers. Ubiquitous Comput. **19**, 259–270 (2015). https://doi.org/10.1007/s00779-014-0820-1
21. Scalabrin, M., Gadaleta, M., Bonetto, R., Rossi, M.: A bayesian forecasting and anomaly detection framework for vehicular monitoring networks. IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP), 1–6 (2017). https://doi.org/10.1109/MLSP.2017.8168151
22. Shigeru, M., Ueno, K., Nishikawa, T.: dlstm: a new approach for anomaly detection using deep learning with delayed prediction. Int. J. Data Sci. Anal. **8**, 137–164 (2019). https://doi.org/10.1007/s41060-019-00186-0
23. Min, H., Xiaowei, F., Zhiwei, J., Ke, Y., Shengchen, Z.: A novel computational approach for discord search with local recurrence rates in multivariate time series. Info. Sci. **477**, 220–233 (2019). https://doi.org/10.1016/j.ins.2018.10.047
24. Zhiwei, J., Jiaheng, G., Jiarui, F.: A novel deep learning approach for anomaly detection of time series data. Sci. Program. **2021**, 1–11 (2021). https://doi.org/10.1155/2021/6636270
25. Xie, M., Hu, J., Han, S., Chen, H.: Scalable hypergrid k-nn-based online anomaly detection in wireless sensor networks. IEEE Trans. Parallel Distrib. Syst. **24**, 1661–1670 (2013). https://doi.org/10.1109/TPDS.2012.261
26. Seungmin, L., Gisung, K., Sehun, K.: Self-adaptive and dynamic clustering for online anomaly detection. Expert Syst. Appl. **38**, 14891–14898 (2011). https://doi.org/10.1016/j.eswa.2011.05.058
27. Jongwon, K., Jeongho, C.: An online graph-based anomalous change detection strategy for unsupervised video surveillance. EURASIP J. Image Video Proc **76**, 1–11 (2019). https://doi.org/10.1186/s13640-019-0478-8
28. Shahid, N., Naqui, I., Qaisar, S.B.: One-class support vector machines: analysis of outlier detection for wireless sensor networks in harsh environments. Artif. Intell. Rev. **43**, 515–563 (2015). https://doi.org/10.1007/s10462-013-9395-x
29. Feng, Z., Fu, J., Du, D., Li, F.: A new approach of anomaly detection in wireless sensor networks using support vector data description. Int. J. Distrib. Sensor Netw. (2017). https://doi.org/10.1177/1550147716686161
30. Huan, Z., Wei, C., Li, G.: Outlier detection in wireless sensor networks using model selection-based support vector data descriptions. Sensors **2018**, 12 (2018). https://doi.org/10.3390/s18124328
31. Raut, S.S., Deshmukh, S.N.: Anomaly detection in data with extremely high dimensional space via online oversampling principal component analysis. J. Comput. Eng. IOSR **16**, 67–73 (2014). https://doi.org/10.9790/0661-16376773
32. Lee, Y.-J., Yeh, Y.-R., Wang, Y.C.F.: Anomaly detection via online oversampling principal component analysis. IEEE Trans. Knowl. Data Eng. **25**, 1460–1470 (2013). https://doi.org/10.1109/TKDE.2012.99
33. Han, N., Gao, S., Li, J., Zhang, X., Guo, J.L., Xinming, G. Z.and Jun: Anomaly detection in health data based on deep learning. Proc. Int. Conf. Netw. Infrastruct. Digit. Contenc. (IC-NIDC), 188–192 (2018). https://doi.org/10.1109/ICNIDC.2018.8525737
34. Jia, W., Chen, W.: Robust anomaly detection using supervised relevance neural gas with discriminant analysis. Int. J. Security Appl. **10**, 41–50 (2016)
35. Subba, B., Biswas, S., Karmakar, S.: Intrusion detection systems using linear discriminant analysis and logistic regression. 2015 Annual IEEE India Conference (INDICON), 1–6 (2015). https://doi.org/10.1109/INDICON.2015.7443533

36. Machhamer, R.: Online offline learning for sound-based indoor localization using low-cost hardware. IEEE Access **7**, 155088–156106 (2019). https://doi.org/10.1109/ACCESS.2019.2947581
37. Vrigazova, B.: The proportion for splitting data into training and test set for the bootstrap in classification problems. Bus. Syst. Res. J. **1**, 228–242 (2021)
38. Singh, V., Pencina, M., Einstein, A.J.: Impact of train/test sample regimen on performance estimate stability of machine learning in cardiovascular imaging. Sci. Rep. **11**, 14490 (2021). https://doi.org/10.1038/s41598-021-93651-5
39. Gholamy, A., Kreinovich, V., Kosheleva, O.: Why 70/30 or 80/20 relation between training and testing sets: A pedagogical explanation. J. Intell. Technol. Appl. Stat. **11**(2), 105–111 (2018)
40. Kohonen, T.: Self-organizing maps. Scientific Data (2001)
41. Szabo, F.E.: Hankel matrix. The Linear Algebra Survival Guide, 140–143 (2015). https://doi.org/10.1016/C2012-0-06836-6
42. Lim, K., Phan, Q., Longman, R.: State-space system identification with identified hankel matrix. Department of Mechanical and Aerospace Engineering Technical Report No. 3045, 140–153 (1998)
43. Tian, J., Azarian, M. M.and Pecht: Anomaly detection using self-organizing maps-based k-nearest neighbor algorithm. European Conference of the Prognostic and Healthy Management Society 2014 **2** (2014). https://doi.org/10.36001/phme.2014.v2i1.1554
44. Seber, G.A.F.: Multivariate observations. John Wiley & Sons (1984)
45. Krzanowski, W.J.: Principles of multivariate analysis: A user's perspective. Oxford University Press (1988)
46. Katser, I.e.a.: Skoltech anomaly benchmark (skab) (2020). https://doi.org/10.34740/KAGGLE/DSV/1693952
47. Yahoo: S5-dA Labeled Anomaly Detection Dataset. Yahoo https://webscope.sandbox.yahoo.com/ (2021)
48. Ahmad, S.: Unsupervised real-time anomaly detection for streaming data. Neurocomputing **262**, 134–147 (2017). https://doi.org/10.1016/j.neucom.2017.04.070
49. Singh, N., Olinsky, C.: Demystifying numenta anomaly benchmark. International Joint Conference on Neural Networks (IJCNN), 1570–1577 (2017). https://doi.org/10.1109/IJCNN.2017.7966038
50. Däubener, S., Schmitt, S., Wang, H., Bäck, K. T.and Peter: Large anomaly detection in univariate time series: An empirical comparison of machine learning algorithms. 19th Industrial conference on data mining ICDM 2019 (2019)
51. Carlos, P., Pinto, R., Gonçalves, G.: Towards bio-inspired anomaly detection using the cursory dendritic cell algorithm. Algorithms **15**, 1 (2022). https://doi.org/10.3390/a15010001