Smart algorithm for unhealthy behavior detection in health parameters

Leonardo F. da Costa*†, Rodrigo T. de Melo*†, Lucas V. Alves*†, Cleilton L. Rocha*, Eriko W. de O. Araujo*,
Gustavo A. L. de Campos*†, Jerffeson T. de Souza*†,
Andreas Triantafyllidis‡, Anastasios Alexiadis‡, Konstantinos Votis‡, Dimitrios Tzovaras‡

*Atlantic Institute, Fortaleza, Brazil

†Graduate Program in Computer Science, State University of Ceara, Fortaleza, Brazil

‡Information Technologies Institute, Centre for Research and Technology Hellas, Thessaloniki, Greece
E-mails: leonardo.costa@aluno.uece.br, {rodrigo_melo, lucas_alves, cleilton_rocha, eriko}atlantico.com.br,
{gustavo.campos, jerffeson.souza}@uece.br, {atriand, talex, kvotis, Dimitrios.Tzovaras}@iti.gr

Abstract—**Obesity is one of the most significant public health problems of the 21st century, having been recognized by the World Health Organization as the epidemic of this century. This problem has been affecting men, women, and children of all races and all ages, particularly in urban areas. OCARIoT is a research and development project financed by the Rede Nacional de Pesquisa (Brazil) and the European Union to conceive a technological solution based on the Internet of Things to face Childhood Obesity. An OCARIoT solution is to use a Decision Support System to encourage children to have healthy habits, with the help of IoT devices. This work presents the development of an algorithm for detecting unhealthy trends and forecast values in time series, which will be a tool to assist the Decision Support System.**

Index Terms—**Childhood obesity, Time series analysis, Forecasting, Unhealthy behavior detection**

I. Introduction

Childhood obesity is reaching alarming proportions in many countries and poses as a global epidemic and serious challenge, being considered one of the biggest health problems of the 21st century. As stated by the World Health Organization, in 2013, an estimated 42 million children under the age of 5 were affected by overweight or obesity. If this trend continues, over 70 million children will be overweight or obese by 2025 [1], [2].

Childhood obesity is a direct cause of many comorbidities in adulthood, including type-2 diabetes [3], hyperlipidaemia [4], non-alcoholic liver disease [5], hypertension [6], [7] and respiratory problems [8].

Among the biological drivers regarding the obesity problem, behavioral factors such as eating habits and sedentary lifestyle are determinant, as shown in many researches [9]–[13].

The early detection of patterns, such as those that indicate that a child is prone to an unhealthy state, may help professionals to build an intervention plan before the child settles at that state.

This work aims to describe the development of an algorithm for automatic trend detection and forecasting in time series as part of a Decision Support System (DSS) for promoting healthy eating habits and a non-sedentary lifestyle for children on the Internet of Things (IoT) ecosystem.

The algorithm is part of the reasoning system of the OCARIoT project [14]. The trend detection algorithm is responsible for detecting where professional intervention is necessary for a specific monitored health parameter. At the same time, the forecasting engine aims to predict the possible future values of the health parameters so that the recommendations can be made regarding the current and future health states.

The main contribution of this work is the use of piecewise linear regression with *overlapping* time windows to estimate the elevation of the trend of data in all data points. With that, we classify aberrant behavior as those who outbound a criterion defined in the next sections. The most recent detection point in the trend (*i.e.*, the last valid detection), serves as the reference point to train a forecasting model.

With that, as soon as an unhealthy trend is detected, an ARIMA forecasting model is fitted using the past data values from this reference point to predict a new one in the future to be used in the DSS.

In the next sections, it is depicted the OCARIoT project itself, its architecture, the detection and forecasting method, the related work and our methodology and results, and for last, we discuss the results obtained so far, and what is expected for future work.

II. The OCARIoT project

The OCARIoT project is an initiative of the Brazilian Ministry of Science, Technology, and Innovation through Rede Nacional de Pesquisa (RNP) and the European Union's HORIZON 2020 Programme. Its objective develops a sophisticated, non-invasive, discrete, and personalized IoT system that can detect and normalize behaviors that put an individual at risk for developing obesity or eating disorders [14]–[18].

The project is partnered by 14 educational, research, and development institutions from Brazil, Portugal, Spain, and Greece. In addition, it includes interdisciplinary integration between health researchers (endocrinologists, nutritionists,

978-1-7281-7303-0/20/$31.00 ©2020 IEEE
DOI 10.1109/COMPSAC48688.2020.0-183

physical educators, psychologists, educators in public health) and Information and Communication Technologies (engineers and computer scientists), aiming at potentiating and promote behavioral changes in health (*e.g.* eating and physical activity).

OCARIoT's main goal is to provide a personalized mobile app coaching solution based on Internet of Things (IoT), Artificial Intelligence, Data Science and gamification strategies that guide children to adopt healthy eating, mental and physical habits in order to promote the improvement of eating habits and physical activity and also the prevention of obesity in children aged from 9 to 12 years.

Through sensors (*e.g.* smart bands), the IoT network will enable observation (collection) of the patterns of childhood daily life activity, health evolution, physiological and behavioral parameters, and environmental data.

All this information, combined with medical standards, will enable us to provide personalized prevention of obesity and/or coping coaching plan that will allow children to remain active and engage in their healthy eating habits and well-being.

III. OCARIoT's architecture

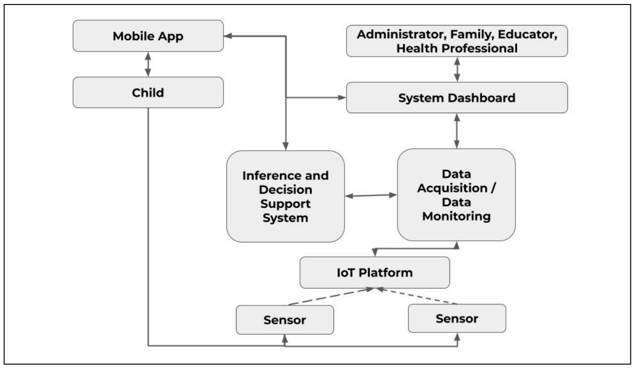
Mobile App
Administrator, Family, Educator, Health Professional
Child
System Dashboard
Inference and Decision Support System
Data Acquisition / Data Monitoring
IoT Platform
Sensor
Sensor

Fig. 1. OCARIoT's architecture overview.

As depicted in Figure 1, the OCARIoT's architecture is composed by many modules, the health parameters are captured by sensors such as smart bands, that measures the number of steps walked in a day, the intervals of sleeping, different kinds of physical activities like cycling, walking, running, jogging, etc.

These data are collected automatically and fed into the IoT platform, responsible for processing the data as message streams for further processing. The platform is also fed with hand-collected questionnaires about children's habits and preferences, anthropometric measurements, parental and medical reports, among others.

The data acquisition/monitoring module is responsible for gathering data from the IoT platform and make the minimal pre-processing steps necessary for posterior analysis in the DSS and inference system. The DSS itself is responsible for learning the relationships among health-related variables and support health professionals to choose a better coaching plan for children.

All the processed data and the coaching plan is available in the dashboard of the application, so the family, educators, and health professionals can follow up on how the children are performing. For the children, a gamified app is available to encourage them to follow the coaching plan.

In this paper, our efforts will be focused specifically in the inference module, since the DSS requires a reference point to be triggered, *i.e*, were correctly to start acting. For that, we analyze the trend of each health parameter, searching for points in time where those tendencies are increasing or decreasing too steeply, and from that point, we establish a forecasting horizon to estimate the health variable state in a near future so that the coaching plans can be recommended regarding the current, past and future states of the health parameters.

IV. Unhealthy trend detection and forecasting

We establish an unhealthy trend, any trend that drives a health parameter out of its safety margins. The World Health Organization well defines those margins.

In Figure 2, we show an example of a time series collected by one smart band; it denotes the amount of walked steps in a daily manner. The dashed line represents the ideal amount of daily steps for children (ten thousand steps daily), and the dotted line represents the minimal amounts of steps to walk daily (five thousand steps). The ideal scenario, in this case, is to stay above the dashed line in all cases.

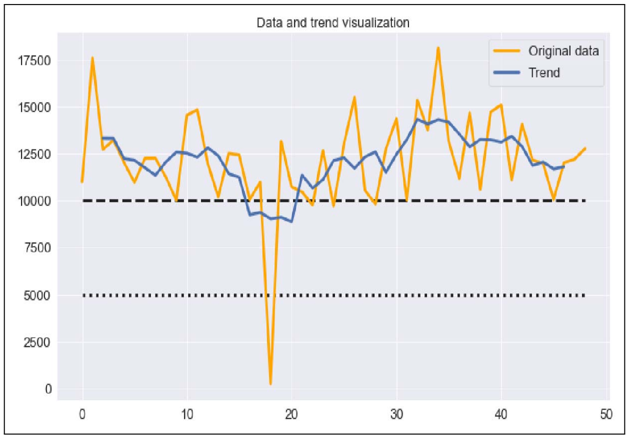

Fig. 2. Example of amount of steps walked daily by a child from OCARIoT project.

The detection of an unhealthy trend resembles the method of piecewise linear regression, except for the fact that it has overlapping sliding windows scanning the signal, as shown in Figure 3. The main goal is to estimate the slopes using linear regression each time the sliding window moves in the trend. It worth noting that we calculate the slopes in the trend of the time series rather than calculating it with the raw data points.

The slope of a function is a factor that can be used to identify these changing trends in time series functions. This value indicates both the direction and steepness of a line. In a linear function $y = mx + b$, the slope is represented by m, which indicates [19]:

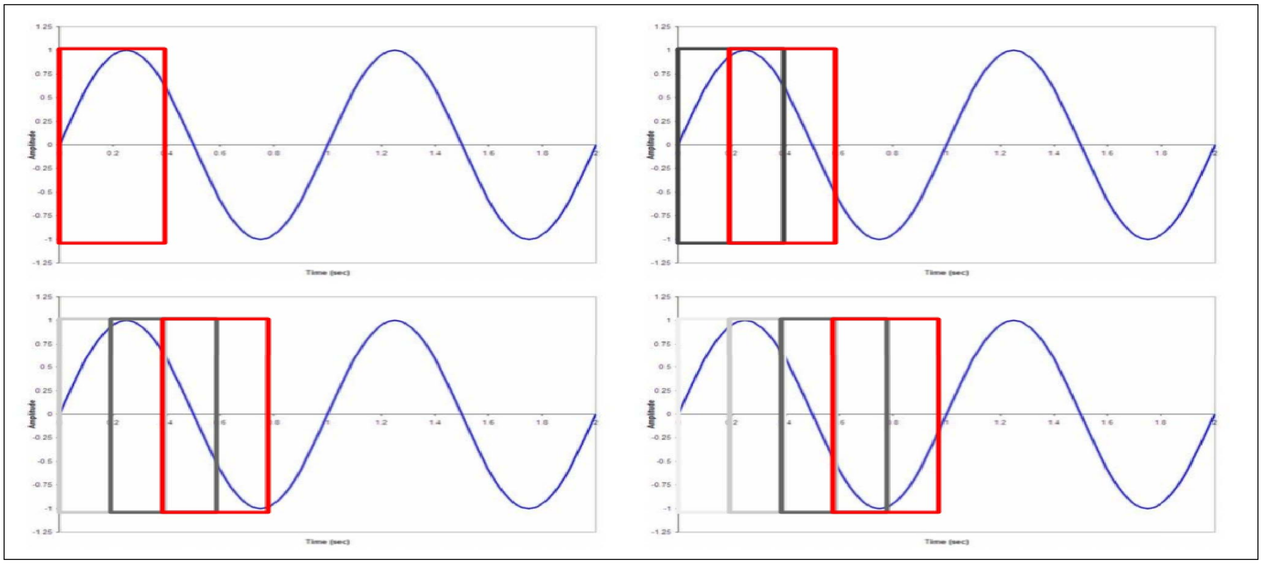

Fig. 3. Example of overlapping sliding window of size 3. In red is the reference window, the faded from black to white windows are already calculated points, the window slides by 1 time step in each figure, thus, overlapping itself.

- $m > 0$ (Positive Slope): The line of the function is increasing if it goes up from left to right (Figure 4);
- $m < 0$ (Negative Slope): The line of the function is decreasing if it goes down from left to right (Figure 5);
- $m = 0$: Is a constant function if the line is horizontal (If the line is vertical, the slope is undefined).

For time series, as they are usually non-linear functions, it is a bit more complicated to find the slope value. It can be found by linear regression with least-squares approximation of the time series values. This method attempts to estimate a linear function from the non-linear function points so that that slope calculation can be possible. The procedure for least squares regression for slope calculation is as follows [20]:

1) For each (x, y) point in the non-linear function to be fitted, calculate x^2 and xy;
2) Sum all x, y, x^2 and xy ($\sum x$, $\sum y$, $\sum x^2$ and $\sum xy$);
3) Calculate Slope m using the equation

$$m = n\frac{\sum xy - \sum x \sum y}{\sum x^2 - (\sum x)^2},$$

where n is the number of points in the function.

Estimating the slopes for each time window, using the overlapping piecewise linear regression, generates a series of slopes $S = [s_{t-T+1}, ..., s_{t-2}, s_{t-1}, s_t]$, $1 \leq t \leq T, \{t, T\} \in \mathbb{Z}$, from the entire series' trend, where T is the length of the time series. For each step t it is calculated the slope for these trend points inside the time window. If the window size is equal to 3, the slope estimation would be done by fitting a linear regression model in those 3 points (Figures 4 and 5), and taking the slope coefficient m from the regression as an approximation of the elevation of this trend.

After getting all slopes of the trend, we first scale the values between -1 and 1, using *min-max scaler* algorithm [21], so the mean (μ) stands as close as possible to zero. We measure μ and the standard deviation (σ) of S, and we establish a sensitivity parameter (ρ) that modulates how much sensible to small variations in trend's elevation the algorithm will be. A unhealthy detection $\bar{y}$ can be of two types: positive or ascending $\bar{y_a}$ and negative or descending $\bar{y_d}$. For the sake of

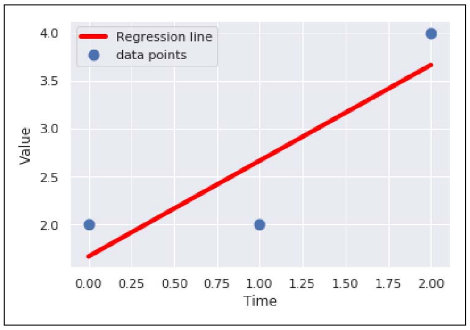

Fig. 4. Example of positive slope ($m > 0$), considering 3 data points.

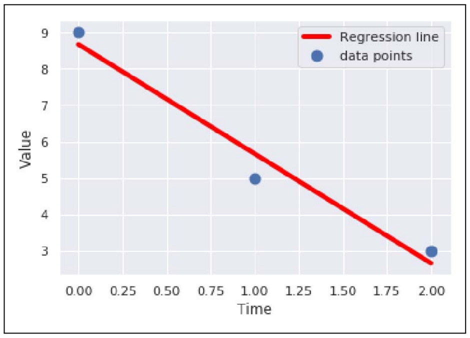

Fig. 5. Example of negative slope ($m < 0$), considering 3 data points.

explanation, we include a third case $\bar{y_n}$ when the slope is not deemed unhealthy (too steep), in the algorithm, this case is ignored, as shown in Equation 1:

$$s_i \in S = \begin{cases} \bar{y_a}, & \textbf{if } s_i > \mu + \sigma\rho \\ \bar{y_d}, & \textbf{if } s_i < \mu - \sigma\rho \\ \bar{y_n}, & \textbf{otherwise}, \end{cases} \tag{1}$$

where $i = \{1, 2, ..., t\}$ and $i \in \mathbb{Z}$.

The result of this algorithm for a time series is shown in Figure 6. The red triangles show the points where the trend is falling too steeply, and the orange triangle shows where the trend is rising too steeply, the last point detected will be used as a reference point to train the forecasting model, described in the next section.

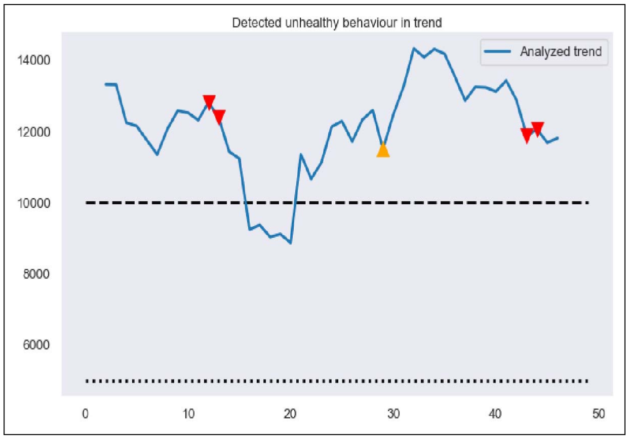

Fig. 6. Analysed trend of daily walked steps of a child in OCARIoT's data, using a sliding window size of 5 and a $\rho = 0.7$, the red triangles denotes where a descending trend is detected, and the yellow triangles denotes the ascending ones.

TABLE I
VALUES DETECTED AS UNHEALTHY TRENDS IN FIGURE 6.

order #	x_value	slope_value
0	0	3781.10
1	1	2322.66
2	12	-830,22
3	13	823.22
4	28	702.12
5	43	-2400.28
6	44	-3580.92

The Table I, shows the values detected by the algorithm in Figure 6, where the column "x_value" are the values detected in x-axis and "slope_value" is the amount of steps increasing or decreasing, according to its signal. The detections $0, 1, 5$ and 6 are not considered valid detections since the sliding window has its middle value as the reference point as it slides, null values may be detected in the head and tail of the array.

V. ARIMA FORECASTING MODEL

The Auto-Regressive Integrated Moving Average model (ARIMA) is a statistical model used to analyze and predict time series data [22]. The model consists of the components described in its name, which are: Autoregressive component (AR), Integrated component (I) and Moving Average (MA).

The AR component is a model which predicts from time series using a dependency relationship between a time series observation and a given value from previous observations. It can be defined by the equation:

$$\left(1 - \sum_{i=1}^{p} \phi_i L^i\right) X_t = \epsilon_t,$$

where X_t is the observed point of the time series at time t, $t > 0$ and $t \in \mathbb{Z}$; p is the number of lag observations considered in this model; ϕ is set of p parameters of the model; L is the time lag operator, defined by $L^i X_t = X_{t-i}$, where $i > 0 \in \mathbb{Z}$; and ϵ_t is the error of X_t.

The MA component is a model that uses the dependence between a time series observation and the errors obtained from previous observations to make the prediction. It can be defined by the equation:

$$X_t = \left(1 + \sum_{i=1}^{q} \theta_i B^i\right) \epsilon_t,$$

where q is the size of the moving average sliding window; θ is the of q parameters of the model; and B has the same function as L, but is applied to the error of the lagged value, where $B^i \epsilon_t = \epsilon_{t-i}$.

An essential concept in the analysis of time series, mainly for prediction using linear models (also called classic models), is the stationary of the time series. A time series is said to be stationary if its data oscillates over a constant average, independent of time, with the variance of the fluctuations remaining essentially the same. Seasonal time series, or with linear or exponential trends, are examples of time series with non-stationary behavior. If the series is not stationary, it may be necessary to make it stationary by differentiating it once or twice, that is, by applying first-order or second-order differentiation.

ARIMA component "I" allows the model to treat non-stationary time series by applying the differentiation factor, $(1 - L)^d$, where d is the differentiation factor. Thus, the complete equation of the ARIMA model is:

$$\left(1 - \sum_{i=1}^{p} \phi_i L^i\right) (1 - L)^d X_t = \left(1 + \sum_{i=1}^{q} \theta_i B^i\right) \epsilon_t \tag{2}$$

The default notation used is $ARIMA(p, d, q)$, and these parameters are obtained from the Box-Jenkins method [23].

To identify parameter d, it is necessary to verify how many times the time series needs to be differentiated to become stationary. This can be done by applying unit root tests, such as the Dickey-Fuller test [24], which verifies the time series stationary. For time series differentiation factor $d = 0, 1, 2, ..., n$, some of these tests must be applied once, and the value of d at which the series became stationary is chosen for $ARIMA(p, d, q)$.

The parameters p and q can be determined from analysis of the autocorrelation and partial autocorrelation charts of the time series analyzed [25], [26].

The next step is to estimate the $\phi = \{\phi_1, \phi_2, ..., \phi_p\}$ and $\theta = \{\theta_1, \theta_2, ..., \theta_q\}$ parameters. These values can be estimated using the least squares method, which starts from the principle of choosing the coefficients so that the sum of the square errors is minimized [27]. So the values of ϕ are chosen in a way that minimizes

$$\sum_{t=1}^{T} \epsilon_t^2 = \sum_{t=1}^{T} (X_t - \phi_1 L_t^1 - \phi_2 L_t^2 - ... - \phi_p L_t^p)^2,$$

where $1 \leq t \leq T, \{t, T\} \in \mathbb{Z}$ and T is the length of the time series.

The θ values are estimated in the same way:

$$\sum_{t=1}^{T} \epsilon_t^2 = \sum_{t=1}^{T} (X_t - \theta_1 L_t^1 - \theta_2 L_t^2 - ... - \theta_q L_t^q)^2.$$

A. Forecasting with ARIMA

After formalizing ARIMA, and describing the methods to identify the best parameters for the model, the next step is to visualize how the prediction is actually made. To do this, consider the following example: Be an $ARIMA(2, 1, 1)$, with estimated parameters $\phi = \{\phi_1, \phi_2\}$ and $\theta = \{\theta_1\}$. So, this equation is:

$$(1 - \phi_1 L^1 - \phi_2 L^2)(1 - L)^1 X_t = (1 + \theta_1 B^1)\epsilon_t \qquad (3)$$

Developing the expression in the Equation 3:

$$[1 - (1 + \phi_1)L + (\phi_1 - \phi_2)L^2 + \phi_2 L^3]X_t = (1 + \theta_1 B)\epsilon_t \quad (4)$$

$$X_t - (1 + \phi_1)LX_t + (\phi_1 - \phi_2)L^2 X_t + \phi_2 L^3 X_t = \epsilon_t + \theta_1 B\epsilon_t \quad (5)$$

Replacing the lag operators L and B in the Equation 5:

$$X_t - (1 + \phi_1)X_{t-1} + (\phi_1 - \phi_2)X_{t-2} + \phi_2 X_{t-3} = \epsilon_t + \theta_1 \epsilon_{t-1} \quad (6)$$

Isolating X_t in the expression in the Equation 6:

$$X_t = (1 + \phi_1)X_{t-1} - (\phi_1 - \phi_2)X_{t-2} - \phi_2 X_{t-3} + \epsilon_t + \theta_1 \epsilon_{t-1} \quad (7)$$

The equation (6) is used to calculate the time series predictions using the $ARIMA(2, 1, 1)$. To calculate the first value, replace t with $T + 1$, since T is the length of the time series. Thus, to calculate the first predicted value X_{T+1} the equation would be:

$$X_{T+1} = (1+\phi_1)X_T - (\phi_1 - \phi_2)X_{T-1} - \phi_2 X_{T-2} + \epsilon_{T+1} + \theta_1 \epsilon_T \quad (8)$$

Since ϵ_{T+1} is unknown, because X_{T+1} is a value that does not yet exist, the value of ϵ_{T+1} is set to zero. Therefore:

$$X_{T+1} = (1+\phi_1)X_T - (\phi_1 - \phi_2)X_{T-1} - \phi_2 X_{T-2} + \theta_1 \epsilon_T \quad (9)$$

For the next predicted values, the value of ϵ_T will be replaced by the residual error of the previous predicted value. Thus, for X_{T+2} the value of ϵ_T will be equal to zero since $\epsilon_{T+1} = 0$. The equation for calculate X_{T+2} is:

$$X_{T+2} = (1 + \phi_1)X_{T+1} - (\phi_1 - \phi_2)X_T - \phi_2 X_{T-1} \quad (10)$$

Generalizing to n predicted values of $ARIMA(2, 1, 1)$:

$$X_{T+n} = (1+\phi_1)X_{T+n-1} - (\phi_1 - \phi_2)X_{T+n-2} - \phi_2 X_{T+n-3} \quad (11)$$

Summarizing the steps of the example above and generalizing for each and every ARIMA configuration, we have the following sequence of actions to forecasting points for time series:

1) Expand ARIMA equation and isolate X_t;
2) Replace t with $T+i$, $1 \leq i \leq n$ and $i, n \in \mathbb{Z}$, where i is the step in the future that you want to be predicted from the past values of the time series (starting at $i = 1$), and n is the amount of future values that are to be predicted with ARIMA;
3) Correct the equation according to the $T+i$ value, setting the residual error value (ϵ) of the predicted values equal to zero, and calculate X_{T+i};
4) Repeat step 2 until the number of n values is predicted by the model.

VI. Related work

Detection of anomalous behavior in time series from the trend component analysis is essential for several applications, and there are several ways to do it, as shown in the papers [28]–[32].

One of the most relevant references regarding the detection of abnormal behaviors in time series trends in the work of [28]. This work defines a mathematical model for automatic time series behavior detection for network monitoring, so that it is possible to monitor multiple network services, and that inconsistencies can be detected automatically.

The work of [29] uses methods for detecting abnormal health trends, applying a piecewise linear regression on the time series trend of longitudinal physiological measurements. The abnormal behavior detected would be the trigger for DSS from data from multiple remote access patients.

In [30] work, an innovative technique was developed for automatic detection of long-term abnormalities in data saved in the cloud. The developed technique applies statistical learning that detects anomalies in the trend component of the time series formed by the analyzed data. It uses piecewise approximation in a similar way to [29] work, combined with statistical metrics such as median and absolute deviation from the median, to check the underlying long-term trend and detect anomalous points in it.

Differently from the previous works mentioned in this section, [31] work uses machine learning to detect abnormal behaviors in time series. They proposed a Long Short Term Memory (LSTM) neural network based on an encoding and decoding scheme (EncDec-AD), which learns to recognize the regular behavior pattern of the analyzed time series and then uses reconstruction error to detect the abnormalities.

Furthermore, [32] work applies the detection of abnormal trends to verify sudden changes in time series of electricity consumption. They applied a model based on dynamic sliding window backtracking to find unusual trending situations that occur for windows of different sizes.

VII. Methodology

This section describes the assembly process of the presented methods in the proposed unhealthy behavior detection algorithm. The dataset used for the experiments is presented, and the processes for testing the proposed algorithm.

A. Dataset

The data is continuously collected by OCARIoT's partner countries, especially in Greece, Brazil, and Spain, from children aged 9 to 12 years. Since the data is still under collection, we use a subset of samples that were already validated for use (there is a need for a thorough anonymization process before using the data).

Twenty-four samples and four variables compose this subset. Moreover, all the data used in this work are time series. The variables consist of daily collections of the number of steps, calorie consumption, minutes of moderate to vigorous physical activity, and sleep time for each child. Samples that have the best regular collections for each of the variables were chosen, that is, at least one week of uninterrupted collection.

Considering this subset, we still do not have enough data to produce forecasting results; in this regard, more data was generated using the R language library GRATIS [33], [34].

In order to generate data using the GRATIS library, it is necessary to input the features of the time series to be generated. These features are spectral entropy, autocorrelations values, spike, linearity, curvature, and measures of the strength of seasonality and trend [35]–[37].

For the generated series to have the same characteristics as the original time series, the same features must be given as input. Thus, the R language library Tsfeatures [38] was used to extract the features of the respective time series of each of the variables in each sample, in order to use these features as input to generate more data using GRATIS. Thus, more than 2000 samples were generated for each time series.

B. Unhealthy behavior detection algorithm

The algorithm is described in Figure 7, and is composed of the following steps:

1) Input the variables' time series: An array of time-dependent values;
2) Decompose the time series and extract its trend;
3) Extract slopes from time-series' trend using piecewise linear regression;
4) Check if the slopes are deemed unhealthy, using the Equation 1;
5) If so, take the last valid detection as the reference point to start forecasting, and verify if the unhealthy behavior continues in the predicted values;
6) In both cases of the trend being unhealthy, with unhealthy predicted values or unhealthy with non-unhealthy predicted values, these results will be considered by DSS when acting.

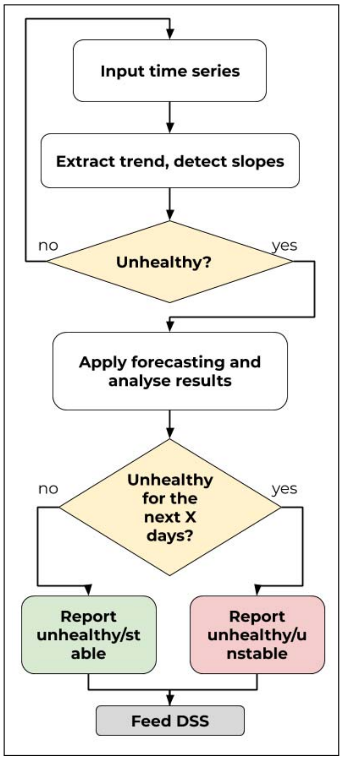
Input time series
Extract trend, detect slopes
no
Unhealthy?
yes
Apply forecasting and analyse results
no
Unhealthy for the next X days?
yes
Report unhealthy/stable
Report unhealthy/unstable
Feed DSS

Fig. 7. Unhealthy behavior detection algorithm.

C. Experiments

The experiments with the method proposed in this work were conducted using 1 of the 24 samples selected from those present in the OCARIoT collections. For the experiments, the health variable chosen to have its time series analyzed by the implemented algorithm was the number of daily steps, as it is a variable whose thresholds for healthy behavior are well defined by the World Health Organization. The experiments were applied to the synthetic data generated from the chosen sample, so each sample variable has a size of 2000 collections.

As for the parameters for the algorithm in the tests:

- The size of the time window was set to 7, to represent one week;
- The sensitivity value ρ was tested using $\rho = \{1, 2, 3\}$.

The sensitivity values were chosen to simply multiply the influence of the standard deviation σ once, twice, and three times, respectively, when determining slopes that are outside the values considered normal. Values greater than three were not chosen because steep slopes would no longer be detected, since all slope values would be within the range considered normal, considering the experiments with the chosen sample.

With the sliding time window of size 7, the fourth backward value was considered as the trigger point for forecasting. The value is obtained considering the idea of the midpoint of the time series explained in Section IV, where 4 is half the value 7 of the sliding time window, rounding upwards.

For experiments, it was considered that the chosen trigger point also indicates the current instant of the analyzed sample, and any value in the original series after this point was considered as future values of the time series.

The ARIMA model used for prediction was ARIMA (5,0,2). All original data from the time series before the time indicated by the trigger point chosen to train the ARIMA model were used. The equivalent of 7 days of data after the trigger point was predicted.

VIII. Results

In this section the results of the experiments with the proposed algorithm are presented. In the Figures 8 and 9, and the Table II show the results using the sensitivity value equal to 1. The Figures 10 and 11, and the Table III represent the results setting the sensitivity value equal to 2. And finally, the results using the sensitivity value equal to 3 are presented in the Figures 12 and 13, and in the Table IV.

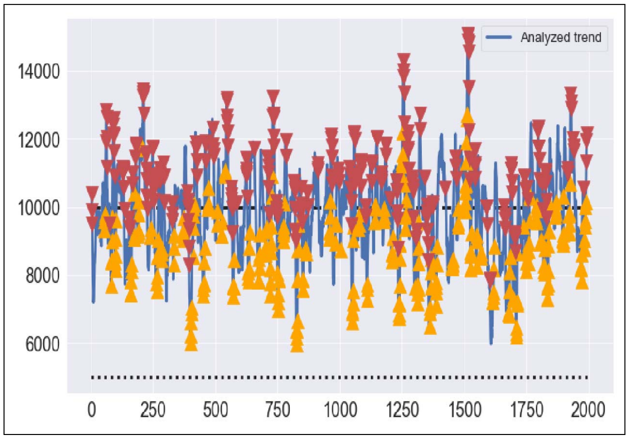

Fig. 8. Analyzed trend of daily walked steps of the data generated from the chosen example, using a sliding window size of 7 and a $\rho = 1$, the red triangles denotes where a descending trend is detected, and the yellow triangles denotes the ascending ones.

In Figure 8, the large number of unhealthy trends identified with a sensitivity value of 1 is notable. The values found are shown in the Table II. Five hundred seventy-eight points classified as possible indicators of unhealthy behavior were found in the trend of the time series of the daily number of steps (Only some of the first and last values found are shown in the table, as there would be no space to show the values in this paper).

TABLE II
Values detected as unhealthy trends in Figure 8.

order #	x_value	slope_value
0	0	1961.464286
1	1	1484.117347
2	2	657.785714
3	3	-479.336735
4	4	-452.122449
...	...	...
573	1988	426.010204
574	1989	346.030612
575	1991	-1225.397959
576	1992	-2158.857143
577	1993	-2600.760204

In theory, considering the variable under analysis, negative growth trends would be the most appropriate to be triggers for the forecast stage, as it is considered sedentary behavior to take less than 5000 steps per day. The ideal for this health variable is to have daily values greater than 10000 steps.

However, the proposed algorithm considers both types of detected unhealthy trends. This happens so that the algorithm has a unified approach for each and every type of health variable. In addition, any trend that can identify too much growth of any health variable, be it positive or negative growth, must be considered, since any exaggeration can be dangerous and must be analyzed.

In this case, the trigger value chosen was number 574 in Table II, which indicates a slight positive growth.

The prediction results are illustrated in Figure 9. For better visualization, only two months of data before the trigger point were illustrated in the figures with the forecasting results.

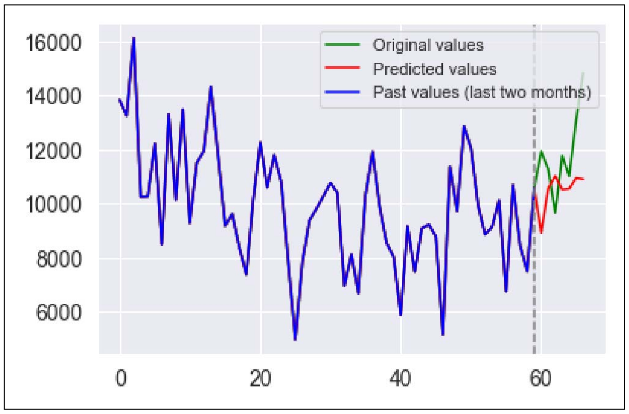

Fig. 9. Prediction results of 7 days after the trigger point chosen from the unhealthy trend points detected by the algorithm using a sliding window size of 7 and a $\rho = 1$.

It is identified behavior of positive growth not very marked of the predicted values, respecting the indicative of the growth of the trigger point. This information would be reported to the DSS to generate the appropriate action plan to correct the unhealthy habits of the individual in this sample.

In Figure 10, as the sensitivity value increased to 2, the number of values classified as unhealthy trends dropped dramatically. It is possible to visualize that the points started to be located preferably in the lower and upper ends of the graph, indicating the refinement of the values found for only those that represent excessive growth and decrease of the time series. Some of these values were shown in Table III.

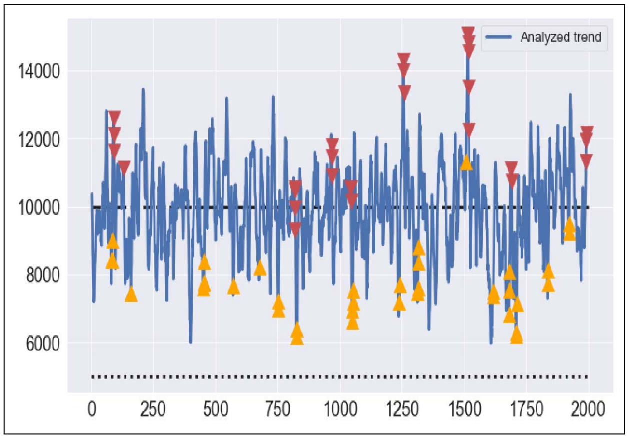

Fig. 10. Analyzed trend of daily walked steps of the data generated from the chosen example, using a sliding window size of 7 and a $\rho = 2$, the red triangles denotes where a descending trend is detected, and the yellow triangles denotes the ascending ones.

TABLE III
VALUES DETECTED AS UNHEALTHY TRENDS IN FIGURE 10.

order #	x_value	slope_value
0	0	1961.464286
1	1	1484.117347
2	2	657.785714
3	83	653.556122
4	84	704.867347
...	...	...
64	1922	690.035714
65	1923	688.688776
66	1991	-1225.397959
67	1992	-2158.857143
68	1993	-2600.760204

In this case, the chosen slope value was number 65 in Table III, which indicates positive growth in the analyzed variable.

The forecasting results are shown in Figure 11. Where again, it is verified that the growth of the predicted values respected the orientation of the unhealthy growth trend value considered as a trigger point.

Finally, Figure 12 shows the steep slopes found with the sensitivity parameter equal to 3. Few unhealthy growth trends were detected, which is expected since only values above the slope average are allowed plus three times the standard deviation, or below the minus average three times the standard deviation. All slope values found for this configuration are shown in Table IV.

For this scenario, the slope in position 2 was chosen as the trigger point, which indicates a negative growth in the time series from it.

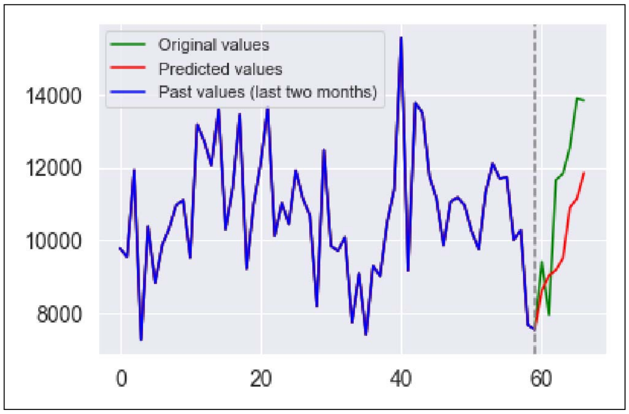

Fig. 11. Prediction results of 7 days after the trigger point chosen from the unhealthy trend points detected by the algorithm using a sliding window size of 7 and a $\rho = 2$.

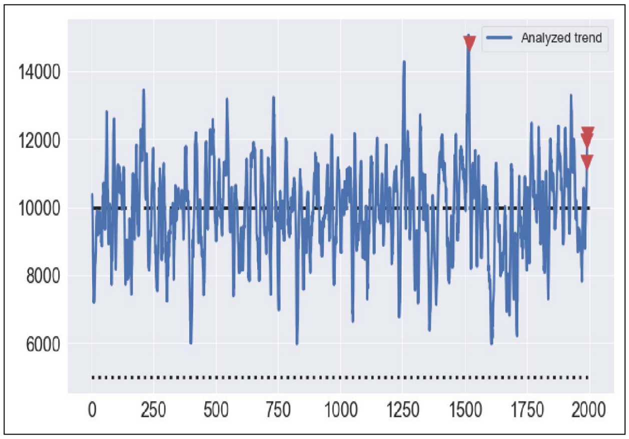

Fig. 12. Analyzed trend of daily walked steps of the data generated from the chosen example, using a sliding window size of 7 and a $\rho = 3$, the red triangles denotes where a descending trend is detected, and the yellow triangles denotes the ascending ones.

TABLE IV
VALUES DETECTED AS UNHEALTHY TRENDS IN FIGURE 12.

order #	x_value	slope_value
0	0	1961.464286
1	1	1484.117347
2	1518	-899.964286
3	1991	-1225.397959
4	1992	-2158.857143
5	1993	-2600.760204

Figure 13 shows the results of the 7-day forecasting from the trigger point. The predicted values again corresponded to the expected negative growth trend.

IX. Conclusion and Future Work

In this work, an algorithm for analyzing time series was presented in order to identify unhealthy behaviors in the series of health variables. The method was developed to assist the DSS that will be implemented in the OCARIoT project.

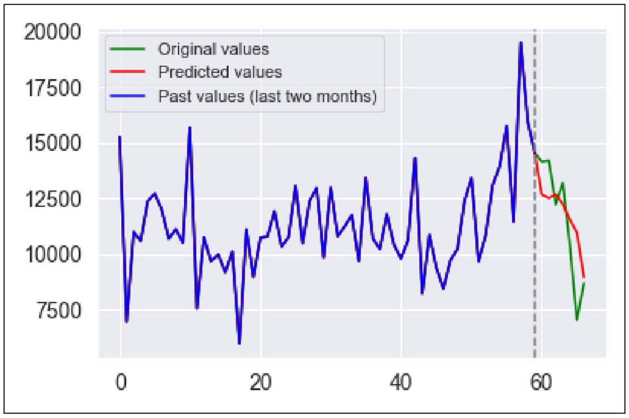

Fig. 13. Prediction results of 7 days after the trigger point chosen from the unhealthy trend points detected by the algorithm using a sliding window size of 7 and a $\rho = 3$.

The algorithm identifies points that indicate excessive growth or decrease in the trend component of the time series of a given health variable, which would indicate unhealthy behaviors.

From the detected points, forecasting of a certain number of points is applied in the future, to check if from the respective point the series behavior remains unhealthy. The information collected will be transmitted to the DSS to identify the appropriate actions for the given situation.

For future work, it is intended to identify other techniques for identifying outliers to eliminate the dependence on the sensitivity parameter. It is intended to benchmark other prediction algorithms to choose which one to use in the proposed algorithm. Furthermore, we want to implement an automated workflow for all stages of the algorithm for detecting unhealthy behavior.

ACKNOWLEDGMENT

The authors would like to thank financial support of the Brazilian National Research and Educational Network (RNP) and the Ministry of Science, Technology, Innovations and Communications (MCTIC) through the 4th Coordinated Call BR-EU in TIC.

REFERENCES

[1] World Health Organization *et al.*, “Interim report of the commission on ending childhood obesity,” World Health Organization, Tech. Rep., 2015.

[2] ——, “Final report of the commission on ending childhood obesity,” World Health Organization, Tech. Rep., 2016.

[3] T. S. Hannon, G. Rao, and S. A. Arslanian, “Childhood obesity and type 2 diabetes mellitus,” *Pediatrics*, vol. 116, no. 2, pp. 473–480, 2005.

[4] A. S. Wierzbicki and A. Viljoen, “Hyperlipidaemia in paediatric patients,” *Drug safety*, vol. 33, no. 2, pp. 115–125, 2010.

[5] T. Reinehr, C. Schmidt, A. M. Toschke, and W. Andler, “Lifestyle intervention in obese children with non-alcoholic fatty liver disease: 2-year follow-up study,” *Archives of disease in childhood*, vol. 94, no. 6, pp. 437–442, 2009.

[6] J. Sorof and S. Daniels, “Obesity hypertension in children: a problem of epidemic proportions,” *Hypertension*, vol. 40, no. 4, pp. 441–447, 2002.

[7] S. R. Daniels, “The consequences of childhood overweight and obesity,” *The future of children*, vol. 16, no. 1, pp. 47–67, 2006.

[8] D. Chapman, G. King, and E. Forno, “Obesity and lung function: From childhood to adulthood,” in *Mechanisms and Manifestations of Obesity in Lung Disease*. Elsevier, 2019, pp. 45–65.

[9] M. Á. Martínez-González, J. A. Martinez, F. Hu, M. Gibney, and J. Kearney, “Physical inactivity, sedentary lifestyle and obesity in the european union,” *International journal of obesity*, vol. 23, no. 11, p. 1192, 1999.

[10] J. E. Manson, P. J. Skerrett, P. Greenland, and T. B. VanItallie, “The escalating pandemics of obesity and sedentary lifestyle: a call to action for clinicians,” *Archives of internal medicine*, vol. 164, no. 3, pp. 249–258, 2004.

[11] S. V. Kurdaningsih, T. Sudargo, and L. Lusmilasari, “Physical activity and sedentary lifestyle towards teenagers’ overweight/obesity status,” *International Journal of Community Medicine and Public Health*, vol. 3, no. 3, pp. 630–635, 2017.

[12] N. Yahia, A. Achkar, A. Abdallah, and S. Rizk, “Eating habits and obesity among lebanese university students,” *Nutrition journal*, vol. 7, no. 1, p. 32, 2008.

[13] A. D. S. da Cruz, A. J. de Oliveira Castro, A. P. do Nascimento Pereira, A. A. R. de Souza, P. R. A. de Amorim, and R. C. Reis, “Eating habits and physical inactivity in children and adolescents with obesity in the admission of the university hospital of the obesity program bettina ferro de souza/habitos alimentares e sedentarismo em criancas e adolescentes com obesidade na admissao do programa de obesidade do hospital universitario bettina ferro de souza,” *Revista Brasileira de Obesidade, Nutrição e Emagrecimento*, vol. 11, no. 61, pp. 39–47, 2017.

[14] OCARIoT Project. Smart childhood obesity caring solution using iot potential. [Online]. Available: https://ocariot.eu/, last accessed on December 2019.

[15] J. E. V. Filho, C. C. P. Brasil, F. Brito, T. Claussen, I. N. Bezerra, L. de Moura, E. Werbet, P. Barbosa, C. Benevides, and R. L. Verde, “Ocariot: Integrando conhecimentos em saúde e tecnologias mhealth, iot e data science no apoio ao enfrentamento da obesidade infantil,” in *XVI Congresso Brasileiro de Informática em Saúde*, 2018.

[16] E. T. Nakamura and S. L. Ribeiro, “A privacy, security, safety, resilience and reliability focused risk assessment in a health iot system: Results from ocariot project,” in *2019 Global IoT Summit (GIoTS)*. IEEE, 2019, pp. 1–6.

[17] A. Triantafyllidis, A. Alexiadis, D. Elmas, K. Votis, and D. Tzovaras, “A social robot-based platform for prevention of childhood obesity,” in *2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)*. IEEE, 2019, pp. 914–917.

[18] L. Bastida, A. Moya, E. Gaeta, J. E. de Vasconcelos Filho, F. Gabler *et al.*, “The power of gamification to learn and promote healthy habits among children,” in *CEUR Workshop Proceedings*. CEUR Workshop Proceedings, 2019.

[19] C. Clapham, *Oxford concise dictionary of mathematics*. Shanghai Foreign Language Education Press, 2001.

[20] R. Chambers and C. Heathcote, “On the estimation of slope and the identification of outliers in linear regression,” *Biometrika*, vol. 68, no. 1, pp. 21–33, 1981.

[21] E. Bisong, “Introduction to scikit-learn,” in *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Springer, 2019, pp. 215–229.

[22] D. Asteriou and S. G. Hall, “Arima models and the box–jenkins methodology,” *Applied Econometrics*, vol. 2, no. 2, pp. 265–286, 2011.

[23] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[24] D. A. Dickey and W. A. Fuller, “Distribution of the estimators for autoregressive time series with a unit root,” *Journal of the American statistical association*, vol. 74, no. 366a, pp. 427–431, 1979.

[25] M. As’ ad, “Finding the best arima model to forecast daily peak electricity demand,” 2012.

[26] P. J. Brockwell and R. A. Davis, *Introduction to time series and forecasting*. springer, 2016.

[27] F. E. Harrell Jr, *Regression modeling strategies: with applications to linear models, logistic and ordinal regression, and survival analysis*. Springer, 2015.

[28] J. D. Brutlag, “Aberrant behavior detection in time series for network monitoring.” in *LISA*, vol. 14, no. 2000, 2000, pp. 139–146.

[29] S. J. Redmond, J. Basilakis, Y. Xie, B. G. Celler, and N. H. Lovell, “Piecewise-linear trend detection in longitudinal physiological measurements,” in *2009 Annual International Conference of the IEEE*

Engineering in Medicine and Biology Society. IEEE, 2009, pp. 3413–3416.

[30] O. Vallis, J. Hochenbaum, and A. Kejariwal, "A novel technique for long-term anomaly detection in the cloud," in *6th {USENIX} Workshop on Hot Topics in Cloud Computing (HotCloud 14)*, 2014.

[31] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proceedings*. Presses universitaires de Louvain, 2015, p. 89.

[32] A. Zhou, L. Zhu, H. Qiu, J. Ding, and W. Rao, "Detection of abnormal trends in electrical data," in *2015 IEEE International Conference on Progress in Informatics and Computing (PIC)*. IEEE, 2015, pp. 247–251.

[33] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2013. [Online]. Available: http://www.R-project.org/

[34] Y. Kang, R. J. Hyndman, and F. Li, "Gratis: Generating time series with diverse and controllable characteristics," *arXiv preprint arXiv:1903.02787*, 2019.

[35] B. D. Fulcher, M. A. Little, and N. S. Jones, "Highly comparative time-series analysis: the empirical structure of time series and their methods," *Journal of the Royal Society Interface*, vol. 10, no. 83, p. 20130048, 2013.

[36] R. J. Hyndman, E. Wang, and N. Laptev, "Large-scale unusual time series detection," in *2015 IEEE international conference on data mining workshop (ICDMW)*. IEEE, 2015, pp. 1616–1619.

[37] Y. Kang, R. J. Hyndman, and K. Smith-Miles, "Visualising forecasting algorithm performance using time series instance spaces," *International Journal of Forecasting*, vol. 33, no. 2, pp. 345–358, 2017.

[38] R. Hyndman, Y. Kang, P. Montero-Manso, T. Talagala, E. Wang, Y. Yang, and M. O'Hara-Wild. Tsfeatures: Time series feature extraction. [Online]. Available: https://pkg.robjhyndman.com/tsfeatures/, last accessed on December 2019.