Detecting Anomalies in Sequential Data Augmented with New Features

Xiangzeng Kong, Yaxin Bi* and David H. Glass

School of Computing
Ulster University at Jordanstown,
Newtownabbey, Northern Ireland, UK, BT37 0QB
Emails: {y.bi, dh.glass}@ulster.ac.uk

Abstract This paper presents a new weighted local outlier factor method for anomaly detection, which is underpinned with three novel components: (1) a piecewise linear representation defined on the basis of the important points that consist of extreme points and additional points; (2) a set of new features which are used to identify anomalies given the new piecewise linear representation; (3) a weighting schema, assigning different weights to different features by accounting for the discriminant power of the features. The underlying idea of the proposed method is to characterize a time series with a set of four features and then discover abnormal changes by taking account of the closeness of any data points augmented with the new features. The comparative experiments demonstrate that the proposed piecewise representation method has performed well in sequential time series data, and the weighted local outlier factor method has achieved better accuracy and RankPower in detecting anomalies from the same data sets in comparison with the conventional local outlier factor, normalized local outlier factor and HOT symbolic aggregate approximation methods.

Key words: Anomaly detection, sequential data, feature extraction, weighted local outlier factor

1. Introduction

Anomaly detection techniques aim to find patterns that do not conform to expected behavior in the data set (Chandola, Banerjee, & Kumar, 2009; Huang, 2013). These patterns are often called anomalies, outliers, abnormal changes, surprises or discords in different contexts, frequently arising in real-world applications such as bioinformatics and finance (Huang, 2013; Chandola, Mithal, & Kumar, 2008; Keogh, Lin, & Fu, 2005). In this paper we present a new anomaly detection method called weighted local outlier factor (WLOF), which is able to extract and weight features in time series.

In the past decades, many anomaly detection methods have been developed in specific application domains, which can be broadly divided into two categories (Beigi, Chang, Ebadollahi, & Verma, 2011): modeling approaches (including rule-based, pat-

* corresponding author.

tern-matching and model-based approaches), which require the prior knowledge of application domains, and data mining approaches (including similarity-based and statistical approaches), which do not require any prior knowledge of application domains. Hadi used a modeling approach based on statistical estimation of the distribution parameters to identify anomalies in multivariate samples (Hadi, 1994). Tandon, et al. used a Parametric Statistical Modeling approach based on association rule mining-based techniques for network intrusion detection (Tandon, & Chan, 2007). Keogh, et al. used distance based approaches to identify the anomalies in time series (Keogh, Lin, & Fu, 2005). Sun, et al. proposed an algorithm to compute the neighbourhood for each node in bipartite graphs using random walk with restarts and graph partitioning and then used the neighbourhood information to identify abnormal nodes (Sun, Qu, Chakrabarti, & Faloutsos, 2005). Some researchers have combined modeling approaches and data-mining approaches to identify the anomalies in data streams. For example, Chandola et al. proposed a framework for modeling categorical data with a desired set of characteristics and a set of separability statistics, which are helpful for understanding the performance of similarity measures for outlier detection (Chandola, Boriah, & Kumar, 2008). In addition, Aydin, et al. proposed a modified kernel-based tracking methods for detecting the anomalies of railway traffic (Aydin, Karakose, Akin 2015), and Jin, et al. proposed a method for detecting bearing anomalies and fault prognosis using the Kalman filter approach (Jin, Sun, Que, Wang, Chow, 2016). Moreover several surveys have also been reported in the literature on outlier detection for different application areas (Hodge, & Austin, 2004; Zhang, Meratnia, & Havinga, 2008; Gupta, Gao, Aggarwal, & Han, 2014).

The nature of anomalies determines which anomaly detection techniques would be applied. According to the suggestions of Chandola, Banerjee, & Kumar (2009), anomalies can be grouped into three categories as follows. (1) Point anomalies: a data instance is considered as anomalous with the rest of the data, such as in the case of credit card fraud. (2) Contextual anomalies: a data instance is anomalous in a specific context, but not otherwise. Contextual anomalies have been investigated in time series data (Weigend, Mangeas, & Srivastava, 1995) and spatial data (Kou, Lu, & Chen, 2006). (3) Collective anomalies: a collection of data instances is anomalous with respect to the entire data set. Collective anomalies can be found, for example, in electrocardiogram data (Keogh, Lin, & Fu, 2005).

In this paper we focus on collective anomalies in different types of sequential data. In order to find the collective anomalies, we need to segment a time series into a set of sub-series of data, i.e. subsequences. Piecewise linear representation (PLR) (Keogh, et al., 2001; Yankov, et al. 2007; Keogh, et al., 2008) is a common feature representation method which has been used to obtain the main features of time series data or data streams. The main idea of the PLR is using the K connective straight lines to represent a time series with length $n(K<<n)$. The advantages of PLR are summarized as follows: 1) a low-dimensional index structure and 2) high computational efficiency (Keogh, et al., 2001; Yan, Fang, Wu, & Ma, 2013). In fact, PLR can obtain higher precision with a larger number of segments, but that would require more computation time. Keogh et al. also proposed a Piecewise Aggregate Approximation (PAA) method for dimensionality reduction in time series data (Keogh, et al., 2001; Keogh, et al., 2008; Palpanas, et al., 2004), which segments a time series using a fixed size window and uses the average value of each sub-segment to collectively represent a time series. Park et al.

used the monotonic sliding windows segmentation algorithm to represent a time series, and demonstrated good results for a smooth time series data (Park, Kim, & Chu, 2001). However, real world data often include a great deal of noise and the number of segments required is often very large. Peng et al. used the Landmark Model to segment a time series through selecting segment points according to their minimum distance/percentage principle which is a smoothing process and is implemented as a linear time algorithm (Peng, Wang, Zhang, & Parker, 2000). Pratt et al. proposed an important point segmentation method that compresses a time series by selecting some of its minima and maxima (Pratt, & Fink, 2002). In this paper we adopt a Piecewise Linear Representation method based on Important Points (PLR_IP).

Given a new representation of time series data, we also need a method for measuring the difference between data objects (instances) embedded in subsequences in order to detect collective anomalies. Therefore, a PLR method can be used to segment a time series into an alternative representation, and distances of the objects within their neighbourhood can be used to find the anomaly. For instance, Ramaswamy et al. used the distance in the k-nearest neighbourhood to rank the outliers (Ramaswamy, Rastogi, & Kyuseok, 2000). Their approach can be used to compute the top *n* outliers. Breunig et al. used a local outlier factor (LOF), whose value depends on how isolated objects are with respect to the surrounding neighbourhood, as a measure for determining outliers (Breunig, Kriegel, Ng, & Sander, 2000). Although that approach can find meaningful outliers, there are two issues with the LOF method. One is that it does not work well for those features with different orders of magnitude as the features with large magnitude will determine the results, whereas the features with smaller magnitude will have little effect. Another is that the LOF method can recognize the anomalies in time series data based on their original values (Breunig, Kriegel, Ng, & Sander, 2000), but when anomalies are interleaved in regular frequency spectrums or other complex anomalies, the LOF is not able to do so.

In order to address these two issues above, we propose the WLOF, in which all selected features will be taken into account in detecting anomalies. Importantly, we propose to construct four features to represent time series data, three of which are defined on the basis of the PLR_IP, representing three different aspects of a time series. First of all, we average the data points in a subsequence that corresponds to a sliding window. The second and third features are defined as the number of important points and the maximum angle of the subsequence, respectively, which are designed mainly for finding anomalies in regular spectrums. Finally, Lin et al used the Symbolic Aggregate Approximation (SAX) method (Lin, Keogh, Lonardi, & Chiu, 2003) to map a time series into a character string like "cbccbaab", every character in the alphabet representing the feature of one segment (Keogh et al., 2006). Similarly, to represent a segment with a feature, we propose a new feature which is the difference between the values of important points in a subsequence and then compute the maximum difference between important points in a sliding window which may cover several segments. This feature represents the maximum change in all the segments involved in a sliding window. Therefore these features constitute a core for the WLOF method to find anomalies in time series data.

After presenting the WLOF method in detail, we then present experimental results to evaluate it. The experiments have been carried out over 17 benchmark datasets and

the comparative analysis against other approaches to demonstrate the effectiveness of the proposed WLOF method in discovering more anomalies within the time series data.

The paper is organized as follows. In Section 2, we introduce the concept of PLR_IP and WLOF. In Section 3 we present the experimental results over 17 data sets which show that the proposed method can find local outliers. In Section 4 we discuss the effect of different parameters. Finally, Section 5 presents conclusions and future work.

2. Methodology

2.1 Notation

2.1.1 Time series and subsequences

Time series or sequential data exist in many real world domains such as commercial, economic, medical, and gene expression data. These domains typically involve large amounts of data and are updated regularly which make it very difficult to detect anomalies directly in the original time series data. Thus, we separate a time series sequence into a set of relatively short subsequences using a sliding window. Firstly, we give some definitions of a time series sequence and subsequences as follows:

Definition 1： Time series

A sequence of pairs, $T = [(Z_1, t_1), (Z_2, t_2), \ldots (Z_n, t_n)]$, $(t_1 < t_2 < \cdots < t_n)$ where Z_i is a data point in a *d*-dimensional data space, and t_i is the time stamp corresponding to the time at which Z_i occurs $(1 \leq i \leq n)$.

Definition 2: Subsequence (Keogh, Lin, & Fu, 2005)

Given a time series $T = [(Z_1, t_1), (Z_2, t_2), \ldots (Z_n, t_n)]$, a subsequence C of T is a sampling of length $m \leq n$ of contiguous position from p, that is, $C_{p,m} = \left[\left(Z_p, t_p\right), \ldots, (Z_{p+m-1}, t_{p+m-1})\right]$ for $1 \leq p \leq n-m+1$. To get a set of subsequences $C_m = \{C_1, C_2, \ldots, C_{n-m+1}\}$, sliding windows can be defined and used, where each subsequence corresponds to a sliding window, where overlap between two adjacent sliding windows can be adjusted on the basis of different applications.

2.1.2 Anomalous features of a subsequence

A subsequence could be anomalous compared with subsequences or contain an anomaly, which can be characterized with various features of the subsequence, such as average value and the maximum difference between values of important points, etc. In this study, four features have been identified. Prior to defining them, we define the extreme points, important points, piecewise linear representation and fitting error.

Definition 3： Extreme points (Yan, Fang, Wu, & Ma, 2013)

Given a 1-dimentional time series, $T = [(Z_1, t_1), (Z_2, t_2), \ldots (Z_n, t_n)]$, if

($Z_i > Z_{i-1}$ and $Z_i > Z_{i+1}$) or if ($Z_i < Z_{i-1}$ and $Z_i < Z_{i+1}$), the point (Z_i, t_i) is an extreme point.

Definition 4: Important points

Extreme points are important features of time series, but sometimes the distance between two neighbouring extreme points is too large, making it difficult to find an anomaly. For this reason, we introduce a concept of important points that consist of extreme points plus additional points identified by the following two step procedure. The first step identifies several extreme points that represent largest distances between extreme points in the data, and the second step ensures that the distance between the neighbouring points is not too large. The set of important points is obtained by a two step procedure below.

Step1: select extreme points as important points. The first and last data points of subsequences are selected as important points. Then suppose that there are L extreme points in $T = [(Z_1, t_1), (Z_2, t_2), \ldots (Z_n, t_n)]$, where $L < n$. For a specified number of important points required g and parameter $\beta \in (0,1)$, if $L \geq \lfloor \beta(g-2) \rfloor$, $\lfloor \beta(g-2) \rfloor$ extreme points are selected as important points iteratively as follows. At each iteration, the data point (Z_r, t_r) is selected where r satisfies:

$$r = \arg\max_{j \in FI} D[Z_j, Z_{i_j}] \quad (1)$$

where FI is the set of subscripts of extreme points that have not yet been selected as important points, D is a distance measure, and (Z_{i_j}, t_{i_j}) is the currently selected important point that is the nearest to (Z_j, t_j). If $L < \lfloor \beta(g-2) \rfloor$, all the extremes are selected as important points. Note that since we aim to find the abnormal change of time series and because the change in time t is uniform, this means that the distance between two adjacent data points at t is the same, we only select the important points based on the Z value.

Step2: select some additional points as important points if necessary. The remaining $g - 2 - \lfloor \beta(g-2) \rfloor$ important points are also selected iteratively as follows. Suppose, $P = [(Z_{i_1}, t_{i_1}), (Z_{i_2}, t_{i_2}), \ldots (Z_{i_l}, t_{i_l})]$, where $t_{i_1}, t_{i_2}, \ldots, t_{i_l}$ is the set of important points which have been selected. At each iteration the data point (Z_h, t_h) is selected, where $t_h = \left\lfloor \frac{t_{i_a} + t_{i_{a+1}}}{2} \right\rfloor$, $Z_h = Z_{t_h}$, a is obtained as follows:

$$a = \arg\max_{1 \leq j \leq l-1} D[Z_{i_j}, Z_{i_{j+1}}] \quad (2)$$

i.e. we identify the largest distance between the currently selected important points. If $L < \lfloor \beta(g-2) \rfloor$, all the extremes are selected as important points and the remaining $g - 2 - L$ important points are obtained using formula (2).

Here we give an illustration of important points. Suppose that Figure 1 shows a sequence of a time series and six important points are required, and $\beta = \frac{1}{2}$. First of all, the beginning point b_1 and end point b_2 are selected as indicated by the yellow circles, and then we need to select two extreme points as important points according to step 1 of definition 4. Firstly, e_1 is selected and then e_2 is selected according to formula (1) as indicated by the red circles. Now we have selected all extreme points with the number given by $\lfloor \beta(g-2) \rfloor = 2, where, g = 6, \beta = \frac{1}{2}$, thus the rest of the extreme points m_1, m_2 and m_3 cannot be selected as important points. With this situation, we need then to select two additional points as important points according to the step 2 of definition 4 to ensure none of the differences are too large. The largest difference in Z values between neighbouring points is between b_1-e_1. So a_1 is selected as an important point, and then a_2 is selected according to formula (2) since after a_1 has been added the largest difference in Z values is between a_1-e_1. Points a_1 and a_2 are indicated by the green circles. Since large differences between points will affect feature extraction, the six important points identified should be more suitable for this purpose.

Definition 5: Piecewise linear representation (PLR) of time series based on important points (Yan, Fang, Wu, & Ma, 2013)

Given a time series , $T = [(Z_1, t_1), (Z_2, t_2), \ldots (Z_n, t_n)]$, where the set of important points is , $T' = [(Z'_1, t'_1), (Z'_2, t'_2), \ldots (Z'_m, t'_m)]$, where $Z'_1 = Z_1, Z'_m = Z_n$ and $m < n$, then a PLR of T can be obtained by first defining a set of functions: $T_l = [f_1, f_2, \ldots, f_{m-1})]$, where f_j represents a linear fitting function between the points (Z'_j, t'_j) and (Z'_{j+1}, t'_{j+1}). The PLR of T is obtained by replacing each point in T with the point from the function f_j corresponding to the same time point. The fitting sequence can be expressed as follows: $T'' = [(Z''_1, t_1), (Z''_2, t_2), \cdots, (Z''_n, t_n)]$. In this paper set T' represents the set of important points, and T'' represents the set of fitting sequences.

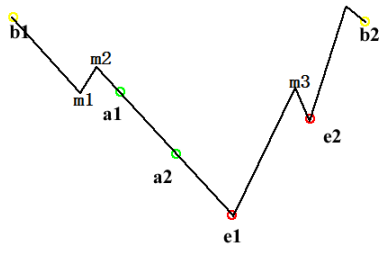


Fig. 1 Illustration of important points

Definition 6: Fitting error of PLR

Having defined the fitting sequence $T^{"}$ which has the same size with original sequence T, the fitting error between the fitting sequence and original sequence T is defined as follows:

$$Err = \sqrt{\sum_{i=1}^{n}(Z_i - Z_i^{"})^2} \tag{3}$$

where n is the length of original sequence, Z_i and $Z_i^{"}$ respectively express the original sequence value and fitting sequence value at the same time t_i. A smaller fitting error shows that the fitting sequence better reflects the original sequence.

According to Definition 5, we develop a segmentation method called PLR_IP, using the important points to segment the time series. Now we further define four features that will be used to characterize subsequences, each of which corresponds to a sliding window, for anomaly detection as follows.

Definition 7: The maximum angle of a subsequence

Let , $T' = [(Z_1', t_1'), (Z_2', t_2'), \ldots (Z_{len}', t_{len}')]$ be the important points in a given subsequence, where len is the number of important points; for simplicity we express this as , $T' = [I_1, I_2, \ldots I_{len})]$. Define θ_i to be the angle between the vectors $V_{i-1,i}$ and $V_{i,i+1}$, where $V_{i-1,i}$ represents the vector from $I_{(i-1)}$ to I_i and $V_{i,i+1}$ represents the vector from I_i to I_{i+1}, for $i = 2,3,\cdots,len-1$. θ_i is called the degree of anomaly of the ith important point as shown in Fig.2. The maximum angle of the subsequences corresponding to a sliding window is denoted S_θ^p and is given by

$$S_\theta^p = \max\left\{|\theta_2|, |\theta_3|, \cdots, |\theta_{len-1}|\right\} \ \theta_i \in (-\pi, \pi) \tag{4}$$

Note that there is no degree of anomaly defined for the first and last important points of a subsequence. The angles are decided by important points; and the fitting data points don't affect the angles.

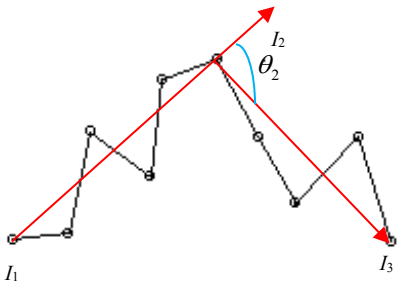


Fig.2 The angle or degree of anomaly of the important point, where I_1, I_2 and I_3 are important points according to definition 4.

Definition 8: Number of important points in a subsequence

The number of the important points in a subsequence, denoted as S_N^p,is defined as

$$S_N^p = \left| \left\{ (Z_\alpha^{'}, t_\alpha^{'}) \in T' \middle| t_p \le t_\alpha^{'} \le t_{p+m-1} \right\} \right| \quad (5)$$

where $T' = [(Z_1', t_1'), (Z_2', t_2'), \ldots (Z_{len}', t_{len}')]$ where $t_1' < t_2' < \ldots < t_{len}'$. T' is a set of important points of the time series T. S_N^p represents the number of important points in C_p computed by Definition 4.

Definition 9: Average value of a subsequence
The average value of Z, denoted as S_μ^p,is defined as

$$S_\mu^p = \frac{1}{m} \sum_{i=p}^{p+m-1} Z_i \quad (6)$$

where p is the beginning position index of the sliding window and $p+m-1$ is the end position index of the sliding window. Z_i represents the value of the data points in the sliding window C_p.

Definition 10: The maximum difference between values of important points in a subsequence

$$S_\sigma^p = \max\{h_2, h_3, \cdots, h_{len}\} \quad (7)$$

where $h_i = |Z_i' - Z_{i-1}'|$ is the difference between $(Z_i^{'}, t_i^{'})$ and $(Z_{i-1}^{'}, t_{i-1}^{'})$ with respect to Z, where $T' = [(Z_1', t_1'), (Z_2', t_2'), \ldots (Z_{len}', t_{len}')]$ are the important points in the sliding window.

2.1.3 A weighted local outlier factor method

According to the features of the time series that have been defined above, we propose a new anomaly detection method called the "weighted local outlier factor", which assigns different features with different weights, and then uses these weighted features for anomaly detection. The relevant definitions are given below.

Definition 11: The distance between two subsequences *P* and *Q* in the new feature space.

We have defined four features in Definitions 7 to 10, which give us a four dimensional feature space. We can then compute the distance between two different subsequences in this space. Supposing subsequence *P* is represented by the point (x_p, y_p, l_p, m_p) and subsequence *Q* by the point (x_q, y_q, l_q, m_q) in the four dimensional feature space, where x, y, l, m represent the four features respectively and the number

of subsequences *n* is determined by the size of the sliding window. The weighted Euclidean distance is defined as follows:

$$\text{wdist}\ (P,Q) \equiv \sqrt{w_1(x_p - x_q)^2 + w_2(y_p - y_q)^2 + w_3(l_p - l_q)^2 + w_4(m_p - m_q)^2} \quad (8)$$

where w_i are weights, which are assigned to these four features and $\sum_{i=1}^{4} w_i = 1$. In order to determine appropriate weights, we use the sum of the values of each feature and want to ensure that for a given feature, the larger its sum, the smaller its weight. This approach can avoid a feature with a large sum determining the result with other features being irrelevant. One way of achieving this is as follows:

$$w_i = \frac{\sum_{j=1}^{4} Sum_j - Sum_i}{3(\sum_{j=1}^{4} Sum_j)} \quad (9)$$

where $Sum_1 = \sum_{k=1}^{n} |x_k|$ for feature *x* and similarly for the other features *y*, *l* and *m*. The idea is that instead of using the normalized sum, i.e. $w_i = \frac{Sum_i}{\sum_{j=1}^{4} Sum_j}$, we use the mean of the normalized sum of the other three features to ensure that the larger sums have the smaller weights. An empirical comparison between the weighted local outlier factor and the local outlier factor is presented in Section 3.

Definition 12: The *k*-distance of subsequence object *P*: $kwdist(P)$ (Breunig et al, 2000)

Here each of the subsequences is viewed as one object which is represented by the four features x, y, l, m. For any positive number *k*, the *k*-distance of object *P*, denoted as *kwdist(P)*, is defined as the *wdist(P,O)* (see definition 11) between *P* and an object $O \in D$, where *D* is the set of subsequence objects such that:

(1) For at least *k* objects $O' \in D \setminus \{P\}$ it holds that $wdist(P,O') \le wdist(P,O)$, and
(2) For at most *k*-1 objects $O' \in D \setminus \{P\}$ it holds that $wdist(P,O') < wdist(P,O)$

These constraints are defined for the *k*-distance of object *P* which represents the distance between *P* and the k^{th} nearest object O. Fig.3 shows the *k*- distance of subsequence object *P*. The definition of the reachability distance of an object is given as follows:

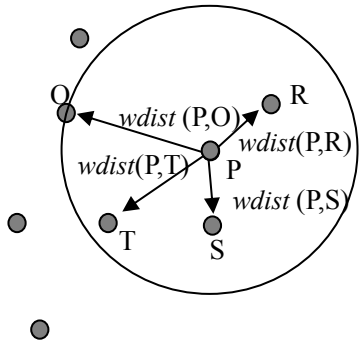


Fig.3 The *k*-distance of subsequence object *P*: $kwdist(p)$ for *k*=4.

Definition 13: The *k*-Weighted local reachability densities of subsequence object P (Breunig et al, 2000):

$$wlrd_k(P) = \frac{k}{\sum_{Q \in kw(P)} reach\text{-}wdist_k(P,Q)} \tag{10}$$

where$kw(P) = \{Q \in D \backslash \{P\}: wdist(P,Q) \leq kwdist(P)\}$,
$reach\text{-}wdist_k(P,Q) = \max\{kwdist(Q), wdist(P,Q)\}$. We can then give the definition of the weighted local outlier factor of an object P based on the reachability distance of an object as follows:

Definition 14: *k*-Weighted local outlier factor of an object *P* (Breunig et al, 2000)

$$WLOF_k(P) = \frac{\frac{1}{k}\sum_{Q \in kw(P)} wlrd_k(Q)}{wlrd_k(P)} \tag{11}$$

According to definition 14, we can get the *k*-weighted local outlier factor of each of the subsequence objects P, and the larger the value of the *k*-weighted outlier factor, the larger the anomaly. From here on this will simply be referred to as the weighted outlier factor, where it is dependent on a constant *k*.

2.2 Anomaly detection algorithm based on weighted local outlier factor

2.2.1 Selection of important points

Based on Definition 4, we present pseudo-code for selecting important points, as shown in Algorithm 1. Therefore, we segment the time series into *g*-1segments using the *g* important points. The description of this method is as follows.

Algorithm: Select important points
Input: The number of important points g and parameter $\beta \in (0,1)$

time series $T=[(Z_1,t_1),(Z_2,t_2),\cdots,(Z_n,t_n)]$

Output: Important points set $T^{'}$

0: Initialise: g; β ; T ; $T^{'}$; FI

1: FI ←Computing the extreme points, $L=|FI|$ L=|FI|(FI is the set of subscripts of extreme points, L is the number of extreme points)

2: $T^{'} \leftarrow (Z_1,t_1)$ and (Z_n,t_n)

If $L \geq \lfloor \beta(g-2) \rfloor$

$numberEP = \lfloor \beta(g-2) \rfloor$; $numberAP = g-2-\lfloor \beta(g-2) \rfloor$

($numberEP$ represents the number of extreme points that need to be selected, $numberAP$ represents the number of additional points that need to be selected)

else

$numberEP = L$;$numberAP = g-2-L$

end

3: *for* $i=1:numberEP$

$T^{'} \leftarrow (Z_r,t_r)$, if $r = \arg\max_{j \in FI} D[Z_j, Z_{i_j}]$

FI ←delete the selected extreme point in FI

i=i+1;

4: **end for**

for $i=1:numberAP$

Compute a according to formula 2

$T^{'} \leftarrow (Z_h,\ t_h)$,the middle of largest segment

i=i+1;

end for

5: Output important points set $T^{'}$

Algorithm 1 Pseudo-code for selecting important points

2.2.2 A new method based on weighted local outlier factor

The proposed anomaly detection algorithm is based on the weighted local outlier factor as shown in Algorithm 2. It involves the following main steps:

Step 1: Uniform scaling. This operation can enlarge or shrink data points by scaling them into the range of 0 and 1.

Step 2: Smooth the data using the locally weighted scatterplot smoothing. In order to find the extreme points, we must smooth the original data set to avoid finding too many extreme points.

Step 3: Selection of important points. We select the important points according to Formula 1 and Formula 2 in Definition 4. The selection of important points is shown in Algorithm 1.

Step 4: Compute the features of subsequences. (1) the maximum angle of the subsequences, (2) the number of important points in the subsequences, (3) the average of the subsequences, and (4) the maximum difference between values of important points of the subsequences.

Step 5: Compute the weighted local outlier factors. Here we compute the weighted local outlier factors of each subsequence based on Definition 14. And then we rank these

weighted local outlier factors and the larger the value of the k-weighted outlier factor, the larger the anomaly.

At the end of the process, the weighted local outlier factor of each subsequence is outputted; the larger values of the weighted outlier factors represent larger anomalies. We will show the sample largest values of the weighted outlier factor of subsequences over different data sets in Section 3.

2.2.3 Metrics for Measurement

Huang (2013) introduced two metrics, which will be used in this study to measure the performance of the anomaly detection algorithms. Suppose the dataset D of n objects contains d_k true anomalies. We use our proposed method to find anomalies that would be ranked within the top 10. Let m_k be the number of true anomalies which are detected by our proposed method in D. Then, we define the accuracy measure of anomaly detection as follows:

$$\text{Accuracy}=\frac{m_k}{d_k} \tag{12}$$

The second measure is called "RankPower" also introduced in (Huang, 2013). Suppose R_i denotes the rank of the ith true anomaly. Then,

$$\text{RankPower}=\frac{m_k(m_k+1)}{2\sum_{i=1}^{m_k} R_i} \tag{13}$$

Larger values of the two metrics imply better performance.

3. Experimental results

Since we are using the sliding window method to obtain the subsequences, we need to set several parameters before conducting an evaluation. We obtain the maximum anomaly values by searching from a minimum value of k=5 to maximum k=20 with a step=1 for the proposed k-weighted local outlier factor method. We use the important points to segment the time series for piecewise linear representation. In Section 3.1 we vary the number of important points to evaluate the effect of the piecewise linear representation, and set it to 10% of the length of the time series in Section 3.2. The sliding window method needs to specify the size of window. Here we set the window sizes to be larger than the time period of the system in time series data in order to find anomalies. We also did the comparison experiments for 50% smaller and 50% larger than our selected window sizes in Section 4. In terms of selecting the extreme points and additional points, we set the parameter β with a value of 1/2.

The experiments start by obtaining the subsequences and selecting important points with the parameter β, by sliding a window of length w across the time series T and then obtaining the features for each of the subsequences, and finally computing the weighted local outlier factor for each subsequence. Note that the index of subsequences goes from 1 to $(n - w) + 1$. The experiments using the piecewise linear representation is

based on important points on the 17 data sets as shown in Table 3, which were downloaded from the website (www.cs.ucr.edu/~eamonn/).

Algorithm: weighted local outlier factor Algorithm

Input: Window size *w*; required number of important points *g* ; smoothing parameter *s*; Times series set *D*

Output: Weighted local outlier factor of data points

0: Initialise: *w,g, m, s,* set of feature values: FV

1: Perform uniform scaling of the times series *D*

2: Smooth the times series *D*

3: Select gimportant points T' according to algorithm1

4: Subdivide the times series *D* into subsequences according to the sliding window size *w*;

5: **for** each subsequence
FV←compute feature values of each subsequence according to the definitions (7)-(10) based on important points T'
end for

6: Compute the weighted local outlier factor for each subsequence based on FV of each subsequence

7: Output weighted local outlier factor of each subsequence

Algorithm 2 Pseudo-code for weighted local outlier factor

3.1 Experimental results of piecewise linear representation based on important points (PLR_IP)

This Section reports the evaluation results on the important points (PLR_IP) to obtain the subsequences. Table 3 presents a summary of some statistics about the 17 data sets used in this work for the comparison between PLR_IP and piecewise linear representation based on the piecewise aggregate approximation (PLR_PAA). In the evaluation, the number of segments over these data sets is determined by the number of important points, from 40 to 100 which is 8% to 20% of the data points for a data set containing 500 points. In the rest of the experiments, we set the number of important points as 10% of the data points. If the length of a data set is larger than 500, we separate the data set into several segments, each of them consisting of 500 data points. If there are less than 500 data points in the last segment, it will be combined with the preceding one as illustrated in Column 3 of Table 3. For example, 500*6+750 in the first row of Table 3, the last segment is 250, which is combined with the previous segment with 500 data points. We then compute the average fitting error. These data sets will then be used to detect anomalies in the following sections. We compute the average fitting error of PLR_IP and average fitting error of PLR_PAA for the different segment numbers (40-100) which is the number of segments of PLR_IP and the number of intervals of PLR_PAA. Fig.4 shows the experimental results for the ECG stdb_308_0 dataset, while the results for all the datasets, which are averaged over the number of segments, are shown in the last two columns in Table 3. We used the t-test to examine the differences between the fitting errors of PLR_IP and PLR_PAA over all the data sets. The single side paired t-test value is 0.22, which indicates that the difference between the PLR_IP and PLR_PAA errors over these data sets is not statistically significant. However, as Table 3 shows, the PLR_IP method indeed gets less fitting error on 9 data sets.

Examining all the data sets, we find that PLR_IP has larger fitting errors for the data sets that have too many peaks such as Space Shuttle Marotta Valve Series and Respiration data set. On the other hand, PLR_IP has smaller fitting errors for data sets with fewer peaks such as Aerospace L-1t and stdb_308_0. Overall, the PLR_IP method can effectively fit these sequential datasets.

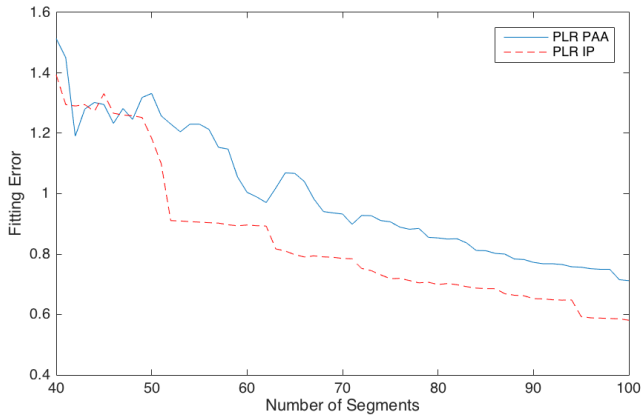


Fig.4 Comparison results for ECG stdb_308_0（1:500）

Table 3 Compared results of fitting error

Number	Data set	Time series length	Type	Error of PLR_IP	Error of PLR_PAA
1	chfdb_chf13_45590	500*6+750	Real	1.3	1.46
2	Lighting2	637	Real	0.72	1.27
3	OliveOil	570	Real	0.67	1.06
4	chfdb_chf01_275	500*6+750	Real	1.30	1.13
5	stdb_308_0 (ECG)	500*9+900	Real	0.92	1.03
6	Respiration	500*8	Real	3.39	2.15
7	Space Shuttle Marotta Valve Series1	500*10	Real	1.31	1.04
8	Space Shuttle Marotta Valve Series2	500*10	Real	1.31	1.02
9	Aerospace L-1q	500*2	Real	1.41	1.8
10	Aerospace L-1b	500*2	Real	6.18	4.95
11	Aerospace L-1j	500*2	Real	6.09	4.89
12	Aerospace L-1p	500*2	Real	1.13	1.8
13	Aerospace L-1t	500*2	Real	1.1	1.33
14	ltstdb_20321_240(ECG)	500*6+750	Real	0.67	1.57
15	xmitdb_x108_0	500*6+750	Real	1.25	0.83
16	respirationppt20	500*3+701	Real	2.44	1.68
17	ltstdb_20321_43(ECG)	500*6+750	Real	1.17	1.79
P value of t-test(all row)		0.22			

3.2 Anomaly detection in electrocardiograms

Electrocardiograms (ECGs) are time series data recording the activities of the heart, which are detected by electrodes attached to the surface of the skin and recorded or displayed by a device external to the body. Given their importance, many annotated data sets have been collected. This experiment has conducted evaluation on three ECG datasets, chfdb_chf01_275, chfdb_chf13_45590 and stdb_308_0 as shown in Figs. 5, 6 and 7, respectively. Fig.5 and Fig.6 are very simple and it is easy to find the anomaly but Fig.7 shows very complicated ECG data where it is difficult to find the anomaly. Fig.5-7 show the original time series (blue line) and the esult using PLR_IP (red line). Table 4 shows the experimental results of the ECG chfdb_chf01_275 using the WLOF and LOF (vector) method which uses the vector of all values of the original subsequence as the input to the LOF method (Breunig et al, 2000), in which the window size is set to w=400 and the number of important points is set to m=375. In this study, we only present the results detected in the top 10 subsequences at most and rank them based on the WLOF values. As seen from Table 4 the strongest outlier is in subsequence 1991. Because the window size is 400, the strongest outlier data point sequence is thus in between 1991-2390, and the second strongest outlier data point sequence is 2163-2560. The rank 1 and rank 2 overlap with the anomaly area as shown by the yellow circle in Fig.5. The anomaly is also detected by the LOF (vector) method in rank 1.

Table 5 shows the results of the ECG chfdb_chf13_45590 using the WLOF and the LOF (vector) method, in which the window size is set to w=250 and important point number is set to m=375. The strongest outlier is subsequence 2728. Because the window size is 250, the strongest outlier data point sequence is in between 2728-2977, in which a possible anomaly area is presented with the yellow circle in Fig. 6. The anomaly is not detected by LOF (vector) method until rank 4. Table 6 shows the results of the ECG stdb_308_0 using the proposed WLOF and LOF (vector) method, with the window size w=400 and important point number m=550. The strongest outlier is in subsequence 1939. Because the window size is 400, the strongest outlier data point sequence is thus in between 1939-2388, and the rank 3 also includes the anomaly area indicated with the yellow circle as shown in Fig.7. The anomaly is detected by the LOF (vector) method in rank 6.

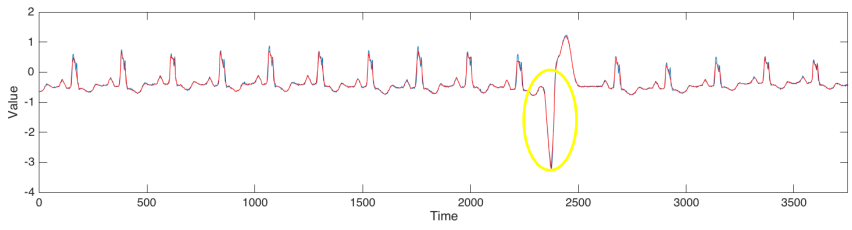


Fig. 5 The time series anomaly found in electrocardiogram chfdb_chf01_275 (marked in yellow circle).

Table 4. Results of the ECG chfdb_chf01_275 for window size = 400

	Rank	1	2	3	4	5	6
WLOFMethod	Subsequence number	1991	2163	1992	2669	2670	2672
LOF(vector) Method	Subsequence number	2388	2663	2389	146	522	315

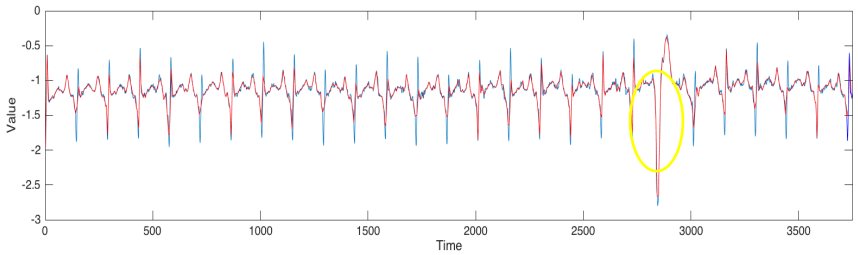


Fig. 6 The time series anomaly found in chfdb_chf13_45590 (marked in yellow circle)

Table 5. Results of the ECG chfdb_chf13_45590 for window size = 250

	Rank	1	2	3	4	5	6
WLOF Method	Subsequence number	2728	2010	865	148	864	2154
LOF(vector) Method	Subsequence number	1151	578	3	2728	3444	2585

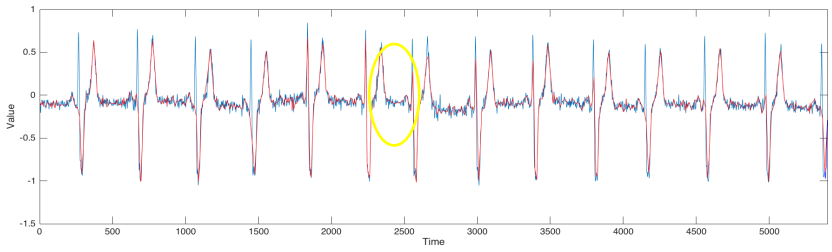


Fig. 7 The time series anomaly found in electrocardiogram stdb_308_0 (marked in yellow circle).

Table 6. Results of the ECG stdb_308_0 with the window size = 400

	Rank	1	2	3	4	5	6
WLOF Method	Subsequence number	1939	3411	2242	3750	3412	4194
LOF(vector) Method	Subsequence number	1505	1418	1386	1393	4485	2544

3.3 Anomaly detection in Space Telemetry

Figs.8 and 9 show two Space ShuttleMarotta Valve series that were annotated by a

NASA engineer (Keogh, Lin, & Fu, 2005). In Fig.8, the expert annotated the anomaly as "Poppet pulled out of the solenoid before energizing". In Fig. 9, the expert annotated the anomaly as "Poppet pulled significantly out of the solenoid before energizing". Tables 7 and 8 show the results of the Space Shuttle Marotta Valve Series 1 and Space Shuttle Marotta Valve Series 2 using the WLOF and LOF (vector) methods, where the window size is set to w=500 and important point number m=500. The strongest outlier subsequence for series 1 according to WLOF starts at 2098 and because the window size is 500 it is therefore the subsequence from 2098-2597, which overlaps with the anomaly area as shown by the yellow circle in Fig. 8. The strongest outlier subsequence for series 2 according to WLOF is 369-868 which does not overlap with the anomaly area as shown by the yellow circle in Fig. 9. However, the 8th strongest outlier subsequence for series 2 is 4030-4529 which does overlap with the anomaly area. Note that none of the subsequences identified by the LOF method in Tables 7 and 8 overlap with the corresponding anomaly areas in Figs. 8 and 9.

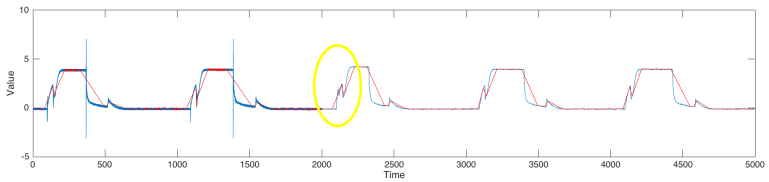


Fig. 8 The time series anomaly found in Space Shuttle Marotta Valve Series 1 (marked in yellow circle)

Table 7. Results of the Space Shuttle Marotta Valve Series 1 for window size = 500

	Rank	1	2	3	4	5	6
WLOF Method	Subsequence number	2098	2594	4234	4233	2595	99
LOF(vector) Method	Subsequence number	515	493	477	479	481	483

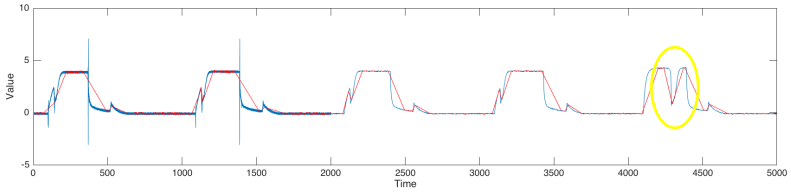


Fig. 9 The time series anomaly found in Space Shuttle Marotta Valve Series 2 (marked in yellow circle)

Table 8. Results of the Space Shuttle Marotta Valve Series 2 for window size = 500

	Rank	1	2	3	4	5	6	7	8
WLOF Method	Subsequence number	369	1091	99	7	596	146	889	4030

LOF (vector) Method	Subsequence number	683	1905	1995	9	13	17	25	35

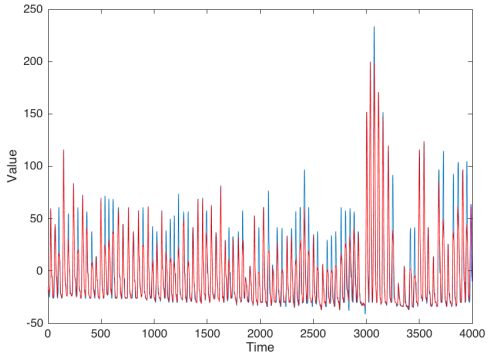


Fig. 10 The original time series of patients' respiration (blue line) and the segmented result (red line).

Table 9. Results of the patients' respiration for window size = 150

	Rank	1	2	3	4	5	6	7
WLOF Method	Subsequence number	2908	2909	2946	702	1262	2947	3390
LOF(vector) Method	Subsequence number	3073	3074	3075	3072	3076	3038	3077

3.4 Anomaly detection in patients' respiration

The Respiration dataset is a time series showing a patient's respiration (measured by thorax extension). The dataset consists of manually segmented data labeled with 'awake' and 'sleep' (Keogh, Lin, & Fu, 2005). Fig.10 shows the original time series of patients' respiration (blue line) and the segmented result (red line). As Fig.10 shows, there are three different stages (0-2950, 2951-3300, and 3301-4000). Table 9 shows the detected results on the Respiration dataset using the WLOF, with the settings of the window size w=150 and important point number m=400. The strongest outliers are subsequences 2908 and 2909, so given the window offset 150, the strongest outlier data subsequence is thus 2908-3057, which includes the change from the first stage to the second stage, and the rank 7 subsequence is 3390-3539 which is just above the change from the second stage to the third stage. The LOF (vector) method finds relevant subsequences in all ranks from 1-7, but they all correspond to the same anomaly from the second stage to the third stage, with the subsequences all being just below the boundary between these stages.

3.5 Anomaly detection in Aerospace data

This section presents the experimental results of the anomaly detection on the Aerospace time series data set (Keogh, Lonardi, & Ratanamahatana, 2004) as shown in Figs. 11-15. Fig.11shows the data set L-1j which is Impulse with one impulse negated inversion. Table 10 shows the results of the AerospaceL-1j, with the window size setting of w=30 and important point number m=100. The strongest outlier is subsequence 480, thus the segment 480-509 overlaps with the anomaly of AerospaceL-1j with one negative impulse as shown in Fig. 11. The anomaly also is detected in rank 1 using the LOF (vector) method. The same parameters for AerospaceL-1b sequence with one impulse amplitude doubled as shown in Fig.12 and Table 11. The strongest outlier is subsequence 471, thus the segment 471-500 overlaps with the anomaly with one impulse amplitude doubled as shown in Fig.12. The anomaly is also detected in rank 1 using the LOF (vector) method.

Fig.13 shows AerospaceL-1p sequence which is the sine with phase advance. Table 12 shows the results of the AerospaceL-1p sequence using the WLOF and LOF (vector) method, where the window size is set to w=30 and important point number m=100.The strongest outlier is subsequence 481, the segment 481-510 overlaps the anomaly of AerospaceL-1p as shown in Fig.13. The LOF (vector) method cannot detect the anomaly in ranks 1-10. Fig.14 shows the AerospaceL-1q sequence which is the sine with phase delay. Table 13 shows the results of the AerospaceL-1q sequence using the WLOF and LOF (vector) methods, with the window size setting of w=30 and segment number m=100. The strongest outlier subsequence according to WLOF is 503-532 which does not overlap with the anomaly area as shown by the yellow circle in Fig. 12. However, the 2nd strongest outlier subsequence is 439-468 which does overlap with the anomaly area. This anomaly is in rank 1 for the LOF (vector) method.

Fig.15 shows the AerospaceL-1q sequence which is the sine with shot noise. The data set has three anomalies with one cycle with a few large magnitude values. Table 14 shows the results of the AerospaceL-1t sequence with the window size setting of w=30 and important point number m=100. The strongest outlier is subsequence 471, the segment is 471-500 which contains one of the anomalies in AerospaceL-1t as shown in Fig.14. Ranks 2, 3 and 4 correspond to the second anomaly and ranks 5 and 6 to the third anomaly. The LOF (vector) method obtains similar results for this data set.

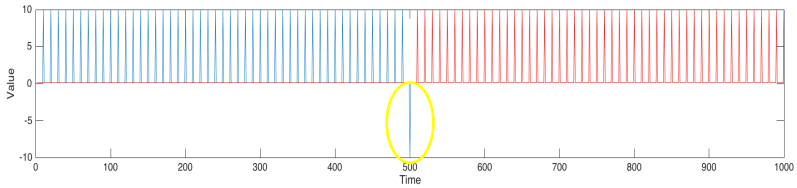


Fig. 11 The time series anomaly found in Aerospace L-1j data (marked in yellow circle)

Table 10. Results of AerospaceL-1j data set for window size = 30

	Rank	1	2	3	4	5	6
WLOF Method	Subsequence number	480	959	960	961	962	963

LOF (vector) Method	Subsequence number	500	498	499	497	9	19

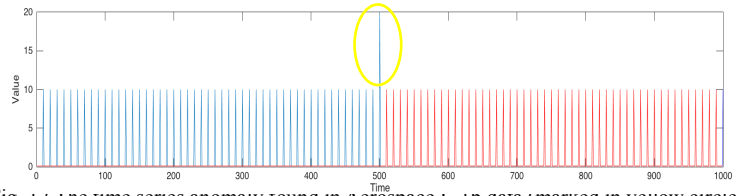


Fig. 12 The time series anomaly found in Aerospace L-1b data (marked in yellow circle)

Table 11. Results of AerospaceL-1b data set for window size = 30

	Rank	1	2	3	4	5	6
WLOF Method	Subsequence number	471	959	960	961	962	963
LOF(vector) Method	Subsequence number	497	7	17	27	37	47

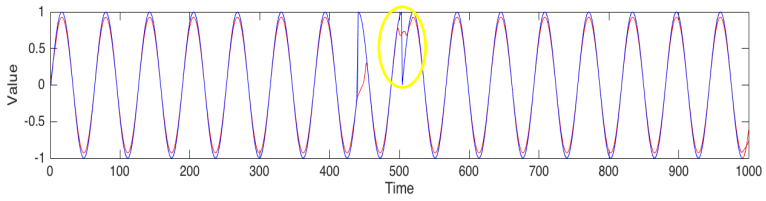


Fig. 13 The time series anomaly found in Aerospace L-1p data (marked in yellow circle)

Table 12. Results of AerospaceL-1p data set for window size = 30

	Rank	1	2	3	4	5	6
WLOF Method	Subsequence number	481	708	645	472	480	648
LOF(vector) Method	Subsequence number	413	916	36	539	350	853

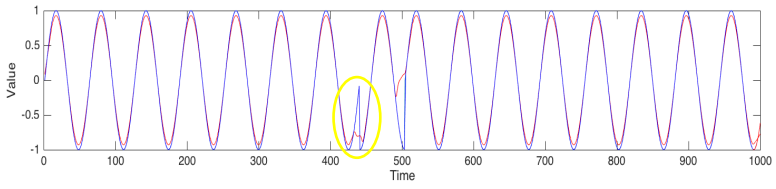


Fig. 14 The time series anomaly found in Aerospace L-1q data (marked in yellow circle)

Table 13. Results of AerospaceL-1q data set for window size = 30

	Rank	1	2	3	4	5	6
WLOF Method	Subsequence number	503	439	444	438	502	440
LOF(vector) Method	Subsequence number	440	439	172	675	109	612

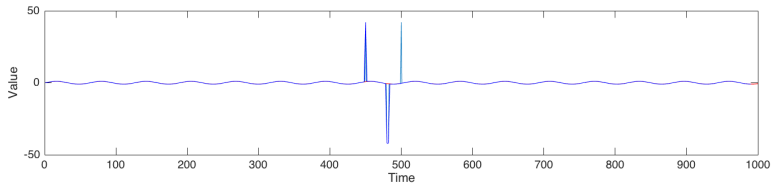


Fig. 15 The time series anomaly found in Aerospace L-1t data

Table 14. Results of AerospaceL-1t data set for window size = 30

	Rank	1	2	3	4	5	6
WLOF Method	Subsequence number	471	482	481	480	421	451
LOF(vector) Method	Subsequence number	480	481	482	500	450	449

The experimental results for the other data sets given in Table 3 are shown in Table 15. There are two anomalies in the Lighting2_TEST data set, which are detected in rank 1 and rank 2. There is only one anomaly for each of the other data sets and the anomalies have been detected in rank 1 in four of the data sets and rank 3 in the other one. The results are compared with method LOF (vector) in Table 16.

Table 15. The experimental results for 5 data sets.

NO.	Data Set	Window size	Number of Segment	Optimal Rank	Beginning point
1	OliveOil_TEST	60	60	Rank 1	451
2	Lighting2_TEST	20	100	Rank 1, Rank 2	457,473
3	respirationppt20	250	200	Rank 1	1530
4	xmitdb_x108_0(ECG)	250	400	Rank 1	4372
5	ltstdb_20321_240(ECG)	100	400	Rank 1	719
6	ltstdb_20321_43(ECG)	100	400	Rank 3	775

4. Discussion

Many rank based anomaly detection algorithms have been proposed such as LOF, Connectivity-based outlier method (COF), and INFLuential measure of outlier by symmetric relationship method (INFLO) (Huang, 2013). They have been used to detect anomalies in several public benchmark data sets. Some anomalies can be detected in rank 1, but they failed to detect some anomalies (Huang, 2013). The empirical results demonstrate that our WLOF method outperforms the LOF method over the seventeen datasets in the different settings of the window and important points. Here we look at the effect

of different parameters. The important point number was set at 10% of the number of data points. We set the parameter window size according to the features of time series data, which should be larger than the length from one peak to next peak in time series data. In order to examine the effect of our feature extraction method, we also obtained results with the LOF method using the features from out method so it could be compared with the LOF method using the vector of all original data points, which was used in section 3.The experimental results are shown in Table 16.

We also examine different window sizes for the WLOF, NLOF, LOF and LOF (vector) methods. The difference between WLOF and NLOF is that instead of constructing four features with different weights, NLOF normalizes the time series data by just mapping each data point into the range [0,1]. The difference between LOF and LOF (vector) is that the input values of the LOF method are the four features of subsequences obtained by our feature extraction method, whereas the input values of LOF (vector) is the vector of all original values of the subsequences. As Table 16 shows, all the anomalies can be detected by the WLOF method using the different window sizes in Section 3, and only one anomaly cannot be detected by the LOF method using our feature extraction method in rank 1-10 as shown in Table 16, however by contrast, 7 anomalies cannot be detected by the LOF (vector) method using the original data point values as the features. Therefore, this result illustrates that our feature extraction method and weighting method have achieved better performance than the LOF methods. For these window sizes, the WLOF can find 100% of the anomalies, the LOF method can find 95% of the anomalies and the LOF (vector) can only find 65% of the anomalies. The WLOF also obtains better rankings for most of these data sets such as data sets 1, 12 and 15, obtaining the best RankPower with a value of (5.12) compared to (3.39) for LOF and (2.76) for LOF(vector). To examine the effect of other window sizes, as Table 16 shows, reducing the window sizes by a half compared to Section 3, 11 anomalies cannot be detected by the LOF (vector) method (just 45% detection rate of the anomalies), 9 anomalies cannot be detected by the LOF and 9 anomalies cannot be detected by the WLOF (55% detection rate), but there are two ranked at 10 by the LOF method. RankPower also can reflect the performance of algorithms; the WLOF obtains a better RankPower (1.83) than the LOF and LOF (vector) methods, which have RankPower of 1.32 and 0.96 respectively. For one and half times the window size in Section 3, 7 anomalies cannot be detected by the LOF (vector) method and LOF method (they find 65% of the anomalies) but 2 anomalies are detected in rank 10 by LOF (vector), and 5 anomalies cannot be detected by WLOF (it finds 75% of the anomalies). And the WLOF also obtains a better RankPower (2.35) than the LOF and LOF (vector) methods, which have RankPower of 2.07 and 2.22 respectively.

We also carried out experiments with NLOF over these datasets. Unlike NLOF, which normalizes the time series, our new weighted method WLOF takes account of the relationship between features by using weights when aggregating all feature together. Table 16 also shows the experimental results obtained using the NLOF, which achieves accuracies of 95%, 55%, and 85% and a RankPower of 2.32, 1.47, and 2.15 for the different windows sizes, respectively. As Table 16 also shows, the WLOF method can obtain accuracies of 100%, 55%, and 75% and RankPower of 5.12, 1.83, and 2.35 for the different windows sizes, respectively. In other words, WLOF can obtain better RankPower than NLOF. As Table 17 shows, the accuracy of finding the anomalies is 100% for β =1/2, 80% for β =2/3 and 75% for β =3/4. These accuracies are better than the results for LOF (vector). Overall, the experimental results demonstrate that our method can

Table 16: Experimental results of different window sizes and methods. Numbers indicate the rank of the subsequence containing the anomaly or two ranks where two anomalies were present. NO indicates that no subsequence containing the anomaly was found in the top 10 ranked subsequences.

No.	data set	Window sizes are set as in Section 3			Window sizes 50% smaller			Window sizes 50% larger		
		WLOF (NLOF)	LOF	LOF(vector)	WLOF (NLOF)	LOF	LOF (vector)	WLOF (NLOF)	LOF	LOF (vector)
1	stdb_308_0	1 (9)	9	NO	1 (1)	1	10	1 (1)	1	10
2	chfdb_chf01_275	1 (1)	1	1	2 (1)	1	10	1 (1)	1	2
3	chfdb_chf13_45590	1 (6)	1	4	NO (5)	NO	NO	4 (NO)	NO	6
4	xmitdb_x108_0(ECG)	1 (2)	2	NO	NO (6)	6	NO	7 (8)	8	2
5	ltstdb_20321_43(ECG)	3 (2)	2	10	NO (NO)	NO	NO	7 (5)	5	NO
6	ltstdb_20321_240(ECG)	1 (2)	2	1	6 (8)	8	NO	3 (4)	4	NO
7	Space Shuttle Marotta Valve Series2	8 (8)	8	NO	NO (NO)	NO	NO	3 (2)	2	NO
8	Space Shuttle Marotta Valve Series1	1 (1)	1	NO	NO (NO)	NO	9	NO (NO)	NO	NO
9	Aerospace L-1t	1,5 (1,5)	1,5	1,5	1,2 (1,NO)	1,NO	1,5	1,6 (1,6)	1,6	1,10
10	Aerospace L-1q	2 (2)	2	1,2	8 (7)	7	9	1 (1)	1	NO
11	Aerospace L-1p	1 (1)	1	NO	1 (1)	1	NO	1 (1)	1	3
12	Aerospace L-1b	1 (10)	6	1	NO (NO)	NO	1	NO (7)	NO	1
13	Aerospace L-1j	1 (8)	1	1	2 (9)	4	NO	7 (7)	NO	1
14	respirationppt20	1 (1)	1	1	1 (1)	1	2	6 (5)	5	1
15	Respiration	1,7 (5,NO)	5,NO	1,NO	NO,NO (NO,NO)	10, NO	(NO,NO)NO, NO	NO (NO)	NO, NO	1, NO
16	OliveOil_TEST	1 (1)	1	3	NO (NO)	NO	NO	NO (7)	NO	NO
17	Lighting2_TEST	1,2 (8,9)	3,4	1,NO	3,9 (NO,5)	10, NO	NO	1,2 (7,8)	3,6	1,2
Accuracy		(95%) 100%	95%	65%	(55%) 55%	55%	45%	(85%) 75%	65%	65%
RankPower		(2.32) 5.12	3.39	2.76	(1.47) 1.83	1.32	0.96	(2.15) 2.35	2.07	2.22

improve the performance of anomaly detection over the 17 data sets with the suitable window sizes in comparison with the LOF methods.

Now we compare our WLOF with the HOT SAX method which was proposed by Keogh, et al. (2005). The authors used their method to represent time series data and then find the discords based on the distance between subsequences. This method also needs to set several parameters. Window size for subsequences is needed and the parameter nseg, which is the number of symbols, is used to represent the subsequence. The element number of the alphabet which is set to 10 in this paper, represents that the HOT SAX method using the alphabet "$a,b,c,\cdots,j$" to represent the subsequence, more details can be found in reference (Keogh, Lin, & Fu, 2005). Table 18 shows the experimental results. The accuracy of all different window sizes is 75% and the RankPower is 2.61 for the window sizes in Section 3 and 2.93 and 4.62 for window sizes 50% smaller and larger respectively. Therefore, the WLOF obtained greater accuracy for the

window sizes set in Section 3 and the same results for window sizes 50% larger, but a lower accuracy for window sizes 50% smaller. While the HOT SAX method has better RankPower results for the smaller and larger window sizes, WLOF obtains the best RankPower (5.12) compared to the other methods for the window sizes set in Section 3 and this is better than any of the results for other methods at any of the window sizes considered.

Table 17 Experimental results for WLOF with different values of the parameter β

		WLOF β =1/2	WLOF β =2/3	WLOF β =3/4
1	stdb_308_0	1	9	NO
2	chfdb_chf01_275	1	1	1
3	chfdb_chf13_45590	1	1	1
4	xmitdb_x108_0(ECG)	1	4	3
5	ltstdb_20321_43(ECG)	3	3	3
6	ltstdb_20321_240(ECG)	1	1	1
7	Space Shuttle Marotta Valve Series2	8	10	NO
8	Space Shuttle Marotta Valve Series1	1	NO	5
9	Aerospace L-1t	1,5	1,2	1,2
10	Aerospace L-1q	2	2	2
11	Aerospace L-1p	1	1	1
12	Aerospace L-1b	1	NO	1
13	Aerospace L-1j	1	NO	NO
14	respirationppt20	1	1	1
15	Respiration	1,7	4,NO	4,NO
16	OliveOil_TEST	1	1	1
17	Lighting2_TEST	1,2	1,2	1,NO
	Accuracy	100%	80%	75%

Table 18, Experimental results of different window sizes and methods

No.	data set	Window sizes are set as in Section 3			Window sizes 50% smaller			Window sizes 50% larger		
		HOT SAX	Window sizes	nseg	HOT SAX	Window sizes	nseg	HOT SAX	Window sizes	nseg
1	stdb_308_0	10	400	40	6	200	20	1	600	60
2	chfdb_chf01_275	1	400	40	1	200	20	1	600	60

3	chfdb_chf13_45590	1	250	25	1	130	13	1	280	28
4	xmitdb_x108_0(ECG)	NO	250	25	NO	130	13	NO	380	38
5	ltstdb_20321_43(ECG)	1	100	25	1	50	5	1	150	30
6	ltstdb_20321_240(ECG)	8	100	25	NO	50	5	NO	150	30
7	Space Shuttle Marotta Valve Series2	1	500	50	1	250	25	1	750	75
8	Space Shuttle Marotta Valve Series1	1	500	50	1	250	25	10	750	75
9	Aerospace L-1t	1,NO	30	10	1,NO	15	5	1,2	45	15
10	Aerospace L-1q	1	30	10	1	15	5	1	45	15
11	Aerospace L-1p	1	30	10	1	15	5	1	45	15
12	Aerospace L-1b	NO	30	10	1	15	5	1	45	15
13	Aerospace L-1j	1	30	10	1	15	5	1	45	15
14	respirationppt20	1	250	25	1	130	13	1	380	38
15	Respiration	1,7	150	15	4,10	80	8	1,2	230	23
16	OliveOil_TEST	NO	30	10	NO	15	5	NO	45	15
17	Lighting2_TEST	10,NO	40	10	10,NO	30	5	NO,NO	60	15
Accuracy/ RankPower		75%, 2.61			75%, 2.93			75%, 4.62		

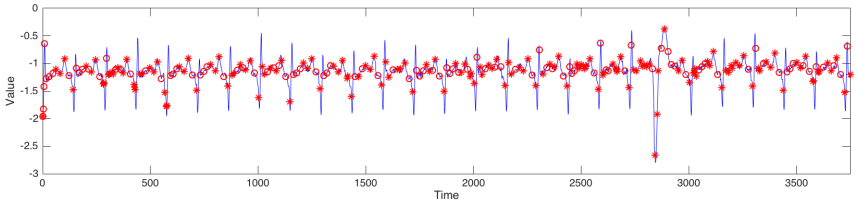
In respect of computational complexity, we compare the WLOF and HOT SAX methods. Suppose n is the size of data sets. Keogh, et al. (2005) have pointed out that the complexity of their method is $O(n^2)$, although they also proposed heuristics to reduce complexity (Keogh, Lin, & Fu 2005), and they later showed an algorithm that can exactly find discords in just $O(n)$ time, with "two linear scans through the database and a limited amount of memory based computation" (Yankov, Keogh & Rebbapragada, 2007). The WLOF and the LOF have the same complexity, but differ from that of HOT SAX. Breunig et al. have analyzed the complexity in (Breunig et al, 2000). The complexity of WLOF and LOF is as follow:

$$T(n) = O(n * t_k) \tag{14}$$

where t_k is the time for a k-nearest neighbour search

For low-dimensional data, the complexity is $O(n)$. For medium to medium high-dimensional data the complexity is $O(n * \log n)$. For extremely high-dimensional data, the complexity is $O(n^2)$.

With respect to the effect of the weighted local outlier factor, Fig. 16 shows the results of important points selection for ECG data set chfdb_chf13_45590 whose parameters are given in Section 3.2. The symbol'*' represents the extreme points and 'o' represents



the additional important points computed by formula 2. As Figure16 shows, the selected important points can segment the time series data and this can help to obtain the four features defined. Table 19 shows the four feature values for the first seven subsequences of chfdb_chf13_45590 and we also find Sum_1=647, Sum_2=77224, Sum_3=3915 and Sum_4=2569, which are obtained as noted after formula (9). Notice that the feature 'the number of important points in the subsequence', Sum_2 is much larger than the other features, which would then dominate the experimental results of the LOF method that uses the four features as input, and so it is unable to find the anomaly in chfdb_chf13_45590 near the data point 2700 as shown in Figure 6. Therefore, we used our WLOF method to allocate different features with different weights. We use the sum of the values of each feature to ensure that for a given feature, appropriate weights are used as given in formula 9. Table 16 shows, for chfdb_chf13_45590, the WLOF method can find the anomaly in rank 1. In summary, the WLOF can make use of all the features in anomaly detection.

Fig. 16 The selected important points of time series ECG chfdb_chf13_45590

Table 19. 4 feature values for the first 7 subsequences of chfdb_chf13_45590

Subsequence number	1	2	3	4	5	6	7
maximum angle	0.28	0.29	0.29	0.28	0.28	0.26	0.27
the number of important points	26	25	24	24	23	23	23
Average value	-1.19	-1.18	-1.17	-1.17	-1.17	-1.17	-1.17
maximum difference between values of important points	0.78	0.78	0.78	0.78	0.78	0.51	0.47

To investigate the discriminability of the four features, we have carried out more experiments on the combinations of these four features, analyzing the effect of combinations of any three features. Table 20 shows the results of any three features. Features 2, 3, 4 can obtain best results with accuracy 100% and RankPower (3.28). However, as shown in Table 16, the addition of feature 1 results in a higher value of RankPower (5.12). The experimental result of combining feature 1, 2, 4 is 0% of accuracy, which indicates that feature 3 "average of the subsequence" is playing a very important role in anomaly detection. Therefore, the use of the four features together can effectively identify the anomalies, but including more features does not necessarily improve the results. For example, in the case of the LOF (vector), where the vector of all values of the original subsequence are used as the input features for the LOF method, the accuracy is low as shown in Table 16.

Table 20. How any tree features affect the results

		Features 1,2,3	Features 2,3,4	Features 1,3,4	Features 1,2,4
1	stdb_308_0	1	5	NO	NO
2	chfdb_chf01_275	2	1	3	NO
3	chfdb_chf13_45590	NO	3	NO	NO
4	xmitdb_x108_0(ECG)	NO	1	2	NO
5	ltstdb_20321_43(ECG)	3	6	10	NO

6	ltstdb_20321_240(ECG)	6	8	NO	NO
7	Space Shuttle Marotta Valve Series2	3	7	3	NO
8	Space Shuttle Marotta Valve Series1	10	1	2	NO
9	Aerospace L-1t	1,2	1,6	1,7	NO
10	Aerospace L-1q	1	4	1	NO
11	Aerospace L-1p	3	1	6	NO
12	Aerospace L-1b	6	1	2	NO
13	Aerospace L-1j	NO	1	NO	NO
14	respirationppt20	2	1	3	NO
15	Respiration	1,3	2,8	1,2	NO
16	OliveOil_TEST	1	1	NO	NO
17	Lighting2_TEST	1,2	2,4	1,4	NO
	Accuracy	85%	100%	75%	0%
	RankPower	3.19	3.28	2.5	0

This section has discussed the experimental results using our WLOF method comparing with LOF, NLOF, LOF (vector) and HOT SAX methods. The experiments show the new features can work better than LOF(vector), and our weighting method can work better than the Normalization method as shown in Tables 16 and 18. The effect of the proposed new features is presented in Table 20. The assessment of different parameter values of β is given in Table 17, with the experimental results demonstrating that the WLOF method can obtain better accuracies than the LOF (vector) for different β values. From all the experiments, it can be found that our important points, features and weighting method can obtain better accuracy and RankPower.

5. Conclusion and future work

In this paper, we have proposed a new WLOF method along with three novel components. The component PLR_IP, which consists of extreme points and additional points, can effectively fit the original time series with the appropriate values of the parameter β. The four features, three of which are defined on the basis of the PLR_IP method, represent different aspects of time series data as the input for the WLOF method. Finally, the weighting schema, which gives the four features with different weights, has made effective use of the discriminant power of all the features together. These novel components effectively characterize the time series data and underpin the WLOF, with the experiments over the seventeen datasets illustrating their effectiveness in anomaly detection.

The comparison between our weighting method and the normalization method demonstrates that the PLR_IP method can effectively extract the features of time series and assist the WLOF method in detecting the anomalies of the time series data. The experimental results also show that the WLOF method can obtain better results over

the 17 data sets than the LOF method, NLOF method, LOF (vector) method and comparable with HOT SAX method. These results indicate that using our feature extraction method can improve the performance of anomaly detection of the LOF method, and our weighting method is better than the normalization method.

One particular issue with the proposed approach is that a number of parameters need to be set prior to the application to anomaly detection. To overcome this shortcoming in practice, we plan to conduct a further study in line with the current research results, including (1) investigating other features for anomaly detection analysis, for example considering the geometrical information of data points; (2) considering a new weighting method which can capture the relationship of all features, and (3) revising the WLOF model to reduce the number of parameters required.

References

Aydin, I. Karakose, M., Akin, E.(2015). Anomaly detection using a modified kernel-based tracking in thepantograph-catenary system. Expert Systems with Applications, 42 (2015) ,938-948

Beigi, M. S., Chang, S. F., Ebadollahi, S., & Verma D. C. (2011). Anomaly detection ininformation streams withoutprior domain knowledge. IBM Journal of Research and Development, 55(5), paper 11, 1-11

Chandola, V., Boriah, S., & Kumar, V. (2008). Understanding categorical similarity measures for outlierdetection. Tech. rep. 08-008, University of Minnesota, 1-45

Chandola, V., Mithal, V., & Kumar, V.(2008). A comparativeevaluation of anomaly detection techniques for sequencedata. ICDM, 743-748

Chandola, V., Banerjee, A. , & Kumar, V., (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), Article 15, 1-58

Edgeworth, F. Y. (1887). On discordant observations. Philosophical Magazine, 23(5), 364–375

Gupta, M. , Gao, J. Aggarwal,C.C. , & Han, J.(2014). Outlier Detection for Temporal Data: A Survey. Knowledge and Data Engineering, IEEE Transactions on, 26(9), 2250- 2267

Hadi, A. S. (1994). A modification of a method for the detectionof outliers in multivariate samples. Journal of the Royal Statistical Society: Series B,56(2) 393 – 396

Hodge, V. J. & Austin, J. (2004). A survey of outlier detection methodologies. Artificial Intelligence Review, 22(2), 85–126

Huang, H. (2013). Rank Based Anomaly Detection Algorithms. Electrical Engineering and Computer Science - Dissertations. 1-182

Jin,X.H.,Sun, Y.;Que,Z.J.;Wang,Y.,Chow,W. S.,(2016).Anomaly Detection and Fault Prognosis for Bearings.IEEE Transactions on Instrumentation and Measurement, 65(9),2046- 2054

Keogh, E., Chakrabarti, K., Pazzani, M. J., & Mehrotra, S. (2008). Dimensionality reduction for fast similarity search in large time series databases. Knowledge and Information Systems, 3(3), 263-268

Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S. (2001). Dimensionality reduction for fast similarity search in large time series databases. Proceedings of the ACM SIGMOD International Conference on Management of Data, pp.151-162

Keogh, E., Lin, J., & Fu, A. (2005). Hot sax: Efficiently findingthe most unusual time series subsequence. ICDM, 226-233

Keogh, E., Lin, J., Lee, S.H., & Herle, H. V. (2006). Finding the most unusual time series subsequence: algorithms and applications. Knowledge and Information Systems, 11(1), 1–27

Keogh, E., Lonardi, S., & Ratanamahatana, C. A. (2004). Towards Parameter-Free Data Mining. KDD, Seattle, Washington, USA, 206-215

Kou, Y., Lu, C.T., & Chen, D. (2006). Spatial weighted outlier detection. In Proceedings of the SIAM Conferenceon Data Mining, 614-617

Lin, J., Keogh, E., Lonardi, S. & Chiu, B. (2003). A symbolic representation of time series, with implications for streaming algorithms. In: Proceedings of the 8th ACM SIGMOD workshop on research issues in data mining and knowledge discovery, 2–11

Breunig , M. M., Kriegel, H-P., Ng, R. N., & Sander, J. (2000).LOF: Identifying Density-Based Local Outliers. Proceeding SIGMOD '00 Proceedings of the 2000 ACM SIGMOD international conference on Management of data, ACM New York, NY, USA, 29 (2), 93-104

Park, S., Kim, S.W., Cho, J.S., & Padmanabhan, S. (2001). Prefix-Querying:An Approach for Effective Subsequence Matching Under Time Warping in Sequence Databases. Proceedings of the 10th International Conference on Information and Knowledge Management, 255-262

Park, S., Kim, S. W., & Chu, W. W. (2001). Segment-Based Approach for Subsequence Searches in Sequence Databases, Proceedings of the 16th ACM Symposium on Applied Computing, 248-252.

Peng, C.S., Wang, H., Zhang, S.R., & Parker, D. S. (2000). Landmarks: A New Model for Similarity-based Pattern Querying in Time Series Databases. Proceedings of the 16th International Conference on Data Engineering, 33-42

Pratt, K. B., & Fink, E. (2002). Search for Patterns in compressed time series. International Journal of Image and Graphics. vol.2(1), 89-106

Ramaswamy, S., Rastogi, R., & Kyuseok, S. (2000). Efficient Algorithmsfor Mining Outliers from Large Data Sets. Proceeding ACMSIGMOD International Conference on Management of Data, 427-438

Sun, J., Qu, H. Chakrabarti, D., & Faloutsos, C. (2005). Neighborhood formation and anomaly detectionin bipartite graphs. In Proceedings of the 5th IEEE International Conference on Data Mining. IEEEComputer Society, 418–425

Tandon, G., & Chan, P. (2007). Weighting versus pruning in rule validation for detecting network and hostanomalies. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discoveryand Data Mining. ACM Press,697-706

Wang H., Fan W., Yu P.S., and Han J.(2003).Mining Concept-Drifting Data Streams

Using Ensemble Classifiers, Proceedings of ACM SIGKDD, 226-235

Weigend, A. S., Mangeas, M., & Srivastava, A. N. (1995). Nonlinear gated experts for time-series: Discoveringregimes and avoiding overfitting. International Journal of Neural Systems, 6(4), 373–399

Yan, C., Fang, J., Wu, L., & Ma, S. (2013). An Approach of Time Series Piecewise LinearRepresentation Based on Local MaximumMinimum and Extremum.Journal of Information & Computational Science 10(9), 2747-2756

Zhang, Y., Meratnia, N., & Havinga, P. J. M. (2008). Outlier detection techniques for wireless sensor networks: A survey. Centre Telemat. Inform. Technol. Univ. Twente, Enschede, The Netherlands, Tech. Rep. TR-CTIT-08-59, 159-170

Dragomir Yankov, Eamonn Keogh, and Umaa Rebbapragada (2007). Disk Aware Discord Discovery: Finding Unusual Time Series in Terabyte Sized Datasets. ICDM 2007

T. Palpanas, M. Vlachos, E. Keogh, D. Gunopulos, W. Truppel (2004). Online Amnesic Approximation of Streaming Time Series. In ICDE . Boston, MA, USA, March 2004